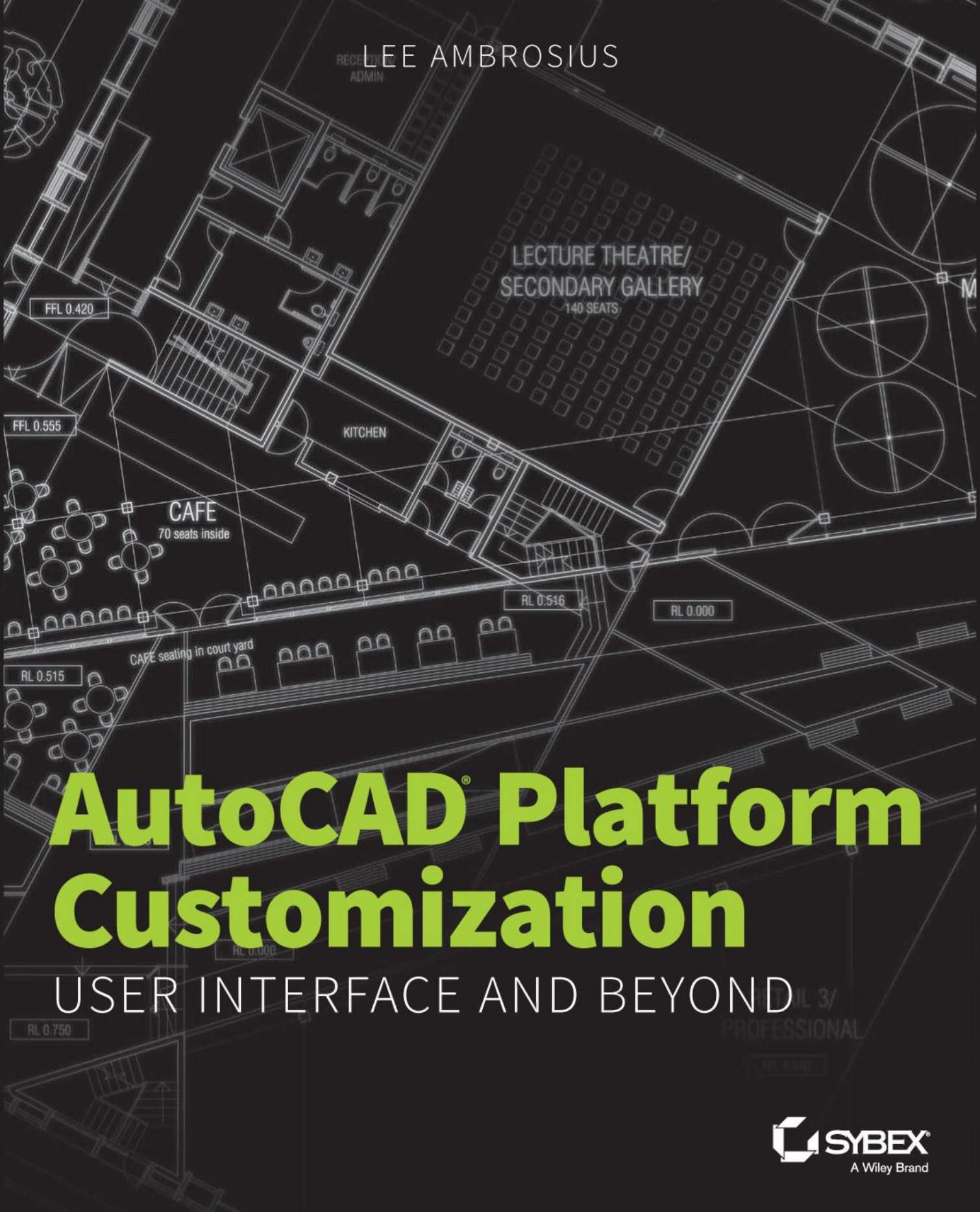


LEE AMBROSIUS

AutoCAD® Platform Customization

USER INTERFACE AND BEYOND

SYBEX®
A Wiley Brand

AutoCAD® Platform Customization

User Interface and Beyond

Lee Ambrosius

SYBEX®
A Wiley Brand

Senior Acquisitions Editor: Willem Knibbe
Development Editor: Mary Ellen Schutz
Technical Editor: Rebecca Afshar
Production Editor: Dassi Zeidel
Copy Editor: Liz Welch
Editorial Manager: Pete Gaughan
Vice President and Executive Group Publisher: Richard Swadley
Associate Publisher: Chris Webb
Book Designers: Maureen Forys, Happenstance Type-O-Rama; Judy Fung
Proofreader: Candace Cunningham
Indexer: Ted Laux
Project Coordinator, Cover: Todd Klemme
Cover Designer: Michael E. Trent/Wiley
Cover Image: © Smileyjoanne/iStockphoto.com

Published simultaneously in Canada
ISBN: 978-1-118-90046-8 (ebk)
ISBN: 978-1-118-90695-8 (ebk)

10 9 8 7 6 5 4 3 2 1

Dear Reader,

Thank you for choosing *AutoCAD Platform Customization: User Interface and Beyond.* This book is part of a family of premium-quality Sybex books, all of which are written by outstanding authors who combine practical experience with a gift for teaching.

Sybex was founded in 1976. More than 30 years later, we're still committed to producing consistently exceptional books. With each of our titles, we're working hard to set a new standard for the industry. From the paper we print on, to the authors we work with, our goal is to bring you the best books available.

I hope you see all that reflected in these pages. I'd be very interested to hear your comments and get your feedback on how we're doing. Feel free to let me know what you think about this or any other Sybex book by sending me an email at contactus@sybex.com. If you think you've found a technical error in this book, please visit `http://sybex.custhelp.com`. Customer feedback is critical to our efforts at Sybex.

Best regards,

Chris Webb
Associate Publisher

Sybex, an Imprint of Wiley

To my wife, who is also my best friend:
It is hard to imagine that I would be writing this book if it were not for you. It was you, all those years ago, who encouraged me to step outside of my comfort zone and share what I knew with others. Thank you for the push I needed and for coming along on this journey with me.

Acknowledgments

I have to give a very special thanks to all the great folks at Sybex for working on and helping to get this project off the ground after a few years of talking about it, especially Willem Knibbe. The next two people I would like to thank are Mary Ellen Schutz and Dassi Zeidel, the developmental and production editors on this book; you two made sure I stayed on track and delivered a quality book. I also want to thank Liz Welch (copyeditor), Candace Cunningham (proofreader), and Ted Laux (indexer) for the work you all did on this book.

Thanks to all the folks at Autodesk, who put in the long hours and are dedicated to the work they do on the Autodesk® AutoCAD® product. Last but not least, I would like to thank Rebecca Afshar, the technical editor on this book. Being a technical editor is never the easiest job, but it is one of the most important and I appreciate what you have done to make this book better.

About the Author

Lee Ambrosius first started working with AutoCAD R12 for DOS in 1994. As a drafter, he quickly discovered that every project included lots of repetition. Lee, not being one to settle for "this is just the way things are," set out on a path that would redefine his career. This new path would lead him into the wondrous world of customization and programming—which you might catch him referring to as "the rabbit hole."

In 1996, Lee began learning the core concepts of customizing the AutoCAD user interface and AutoLISP®. The introduction of VBA in AutoCAD R14 would once again redefine how Lee approached programming solutions for AutoCAD. VBA made it much easier to communicate with external databases and other applications that supported VBA and transformed the way information could be moved between project management and manufacturing systems.

Not being content with VBA, in 1999 Lee attended his first Autodesk University and began to learn ObjectARX®. Autodesk University made a lasting impression on him. In 2001, he started helping as a lab assistant. He began presenting on customizing and programming AutoCAD at the event in 2004. Along the way he learned how to utilize the AutoCAD Managed .NET API.

In 2005, Lee decided cubicle life was no longer for him, so he ventured off into the CAD industry as an independent consultant and programmer for his own company named HyperPics, LLC. After he spent two years as a consultant, Autodesk invited him to work on the AutoCAD team; he has been on the AutoCAD team since 2007. For most of his career at Autodesk, Lee has worked primarily on customization and end-user documentation. Recently, he has been working on the AutoLISP, VBA, ObjectARX, .NET, and JavaScript programming documentation.

In addition to working on the AutoCAD documentation, Lee has been involved with the *AutoCAD and AutoCAD LT Bible, AutoCAD for Dummies, AutoCAD & AutoCAD LT All-in-One Desk Reference for Dummies, AutoCAD 3D Modeling Workbook for Dummies, and Mastering AutoCAD for Mac* books on different editions as a technical writer or author. He has also written white papers on customization for Autodesk and a variety of articles for *AUGIWorld*, published by AUGI®, on customization and programming.

Contents at a Glance

Contents

Introduction

Welcome to *AutoCAD Platform Customization*! Have you ever thought about customizing the Autodesk® AutoCAD® program only to think it is not for you because you're not a programmer? If so, you are not alone—many people connect customization with programming. However, customization is not the same as programming, although programming can be considered a form of customization.

While using one of the supported programming languages can be useful in implementing custom workflows and new commands, simpler ways exist to increase your drafting efficiency in a shorter period of time. AutoCAD supports a wide range of customization features that you can learn and begin to leverage in minutes, which can lead to improved CAD standards and a decrease in the amount of time it takes to complete a task.

I, like many others—even you, most likely—have customized AutoCAD without even realizing it. Have you ever created a new layer, text style, or block? Chances are pretty good that you have created one or more of those items before. You might have even stored those items in a drawing template (DWT) file so they would be available each time a new drawing was created. While you might not have thought about these as forms of customization, they are indeed a few of the basic drawing customization features that can be used to enhance the out-of-the-box AutoCAD experience.

Drawing customization affects the appearance of and settings in a drawing file or drawing template file, and should form the cornerstone of your company's CAD standards. Often when people think of customization, they think of application customization, which includes the support files that AutoCAD uses, as well as the tools in the application's user interface. Application customization is not dependent on which drawing is currently open, but on which user profile or workspace is current.

About This Book

AutoCAD Platform Customization: *User Interface and Beyond* provides you with an understanding of all the customization features to improve your productivity. This book is designed to introduce you not only to concepts related to defining CAD standards and customizing AutoCAD, but also to key customization concepts through procedural-based exercises.

This book is the first book in a series of three that focuses on customizing and programming AutoCAD. The three-book series as a whole is known as *AutoCAD Platform Customization*: *User Interface*, *AutoLISP*, *VBA*, *and Beyond*, and will be available as a printed book in late 2014. Book 2 in the series, *AutoCAD Platform Customization*: *AutoLISP*, will be available in spring 2014; Book 3, *AutoCAD Platform Customization*: *VBA*, will be available in late 2014. These two books

focus on the AutoLISP® and VBA programming languages, starting with the basics and working you toward advanced concepts.

Is This Book for You?

AutoCAD Platform Customization: *User Interface and Beyond* covers many of the customization features that can be found in AutoCAD on Windows and Mac. If any of the following are true, this book will be useful to you:

- You want to learn about which customization options are available in AutoCAD.
- You want to customize the user interface or support files, such as the linetypes and hatch patterns that AutoCAD utilizes.
- You want to automate repetitive tasks.
- You want to create and manage CAD standards for your company.

Customization in AutoCAD

Customization is one of the feature areas that sets AutoCAD apart from many other CAD programs. Even though the product can be used out of the box, configuring the user interface and modifying the support files that come with the product can greatly improve your productivity. By customizing AutoCAD, you can streamline product workflows and create new ones that are a better fit with the way your company works. These workflows might range from importing layers and styles into a drawing to the extraction of drawing-based information into a spreadsheet or database.

Not all of the customization features require you to learn a new tool or skill set; chances are you might have customized AutoCAD and not even realized it. If you have ever created a layer or a block, you already understand some of the customization features of AutoCAD. AutoCAD customization features can be grouped into three categories: basic, intermediate, and advanced. The advanced features require you to know a programming language and an AutoCAD-specific application programming interface (API).

The following outlines many of the common customization and programming options available:

Basic

- Layers
- Annotation styles (text, dimensions, multileaders, and tables)
- Layouts
- Blocks
- Plot styles
- Plotters

- Page setups
- Materials, visual styles, and render presets
- Drawing templates
- Command aliases
- User profiles (Windows only)
- Workspaces (Windows only)
- Desktop icon customization (Windows only)
- Tool palettes (Windows only)

Intermediate

- Scripts
- User interface (CUIx) and DIESEL
- Linetypes and hatch patterns
- Shapes and text styles
- Action macros (Windows only)
- Dynamic blocks (Windows only)

Advanced

- AutoLISP
- ObjectARX
- Visual Basic for Applications (Windows only)
- ActiveX/COM (Windows only)
- Database connectivity (Windows only)
- Sheet Set Manager API (Windows only)
- CAD Standards plug-ins (Windows only)
- Transmittal API (Windows only)
- Managed .NET (Windows only)
- JavaScript (Windows only)

What to Expect

This book is organized to help you learn the fundamentals of establishing CAD standards and then how to manage and implement those standards by customizing the AutoCAD program.

Additional resources and files containing the samples found throughout this book are available on the companion web page, `www.sybex.com/go/autocadcustomization`.

Chapter 1: Establishing the Foundation for Drawing Standards In this chapter, you'll learn how to establish drawing standards. Drawing standards allow you to enforce consistency across multiple drawings. By enforcing a set of standards, you can easily share your drawings and make them look the same when plotting them.

Chapter 2: Working with Nongraphical Objects In this chapter, you'll learn how nongraphical objects affect display and output of objects in a drawing. Nongraphical objects such as layers and text styles make it easy to update the look of all the objects that reference them.

Chapter 3: Building the Real World One Block at a Time In this chapter, you'll learn how to create and manage blocks. Blocks allow you to create logical object groupings that can be used several times in the same drawing. For example, you could create a small assembly of parts and insert it more than once in a drawing. If the assembly changes, you just need to update the block, and all instances of that block are changed.

Chapter 4: Manipulating the Drawing Environment In this chapter, you'll learn how to change the AutoCAD drawing environment. During startup, you can control several of the settings that affect the AutoCAD program. These settings can affect the display of the user interface, behavior of tools in the drawing environment, and where AutoCAD looks for support files.

Chapter 5: Customizing the AutoCAD User Interface for Windows In this chapter, you'll learn how to customize the elements and display of the AutoCAD user interface on Windows. The Customize User Interface (CUI) Editor allows you to create and manage the tools that are displayed by the AutoCAD user interface.

Chapter 6: Customizing the AutoCAD User Interface for Mac In this chapter, you'll learn how to customize the elements and display of the AutoCAD user interface on Mac OS. The Customize dialog box allows you to create and manage the tools displayed by the AutoCAD user interface.

Chapter 7: Creating Tools and Tool Palettes In this chapter, you'll learn how to create and customize tool palettes in AutoCAD on Windows. Tool palettes allow you to create a visual set of tools that can be used to insert blocks, start commands, or even hatch a closed area. Tool palettes are available on Windows only.

Chapter 8: Automating Repetitive Tasks In this chapter, you will learn how to create scripts and action macros to automate repetitive tasks. Script files and action macros allow you to combine multiple commands into simple logical sequences without needing to know a programming language. Action macros are supported on Windows only.

Chapter 9: Defining Shapes, Linetypes, and Hatch Patterns In this chapter, you will learn how to create custom shapes, linetypes, and hatch patterns that you can use to control the way line work appears in a drawing. The AutoCAD install provides a limited number of standard shapes, linetypes, and hatch patterns. You can extend the standard definitions by creating your own shapes, linetypes, and hatch patterns for use in your drawings.

Chapter 10: Using, Loading, and Managing Custom Files In this chapter, you will learn how to use, manage, and migrate custom files. After you have spent the time customizing AutoCAD, all you have left to do is deploy and manage your files.

Companion Website

An online counterpart to this book, the companion website contains the sample files required to complete the exercises found in this book in addition to the sample code and project files used to demonstrate some of the programming concepts explained in this book. In addition to the sample files and code, the website also contains some additional resources that are not mentioned in this book. The companion website can be found at `www.sybex.com/go/autocadcustomization`.

Other Information

This book assumes that you know the basics of your operating system—Windows or Mac OS X—and AutoCAD 2009 or later. When appropriate, I specify when a feature does not apply to a specific operating system or release of AutoCAD. Most of the images in this book were taken using AutoCAD 2014 in Windows 8 and Mac OS X 10.7.

Although this book was not written to cover AutoCAD LT®, much of the content does apply to that software package.

Styles and Conventions of This Book

This book uses a number of styles and character formats—bold, italic, monotype face, all uppercase or lowercase letters, and others—to help you distinguish from the text you read, sample code you can try, text that you need to enter at the AutoCAD Command prompt, or the name of an object class or method in one of the programming languages.

As you read through this book, keep the following conventions in mind:

- User-interface selections are represented by one of the following methods:
  - Click the Application button ➢ Options.
  - On the ribbon, click the Manage tab ➢ Customization ➢ User Interface.
  - On the menu bar, click the Tools menu ➢ Customize ➢ Interface.
  - In the drawing window, right-click and click Options.
- Keyboard input is shown in bold (for example, type **`cui`** and press Enter).
- Prompts that are displayed at the AutoCAD Command prompt are displayed as monospace font (for example, `Specify a start point:`).
- AutoCAD command and AutoLISP function names are displayed in all lowercase letters with a monospace font (for example, `line` or `command`).
- Example code and code statements that appear within a paragraph are displayed in monospace font. Code samples might look like one of the following:
  - `(command "._circle" PAUSE 3)`
  - The `MsgBox` method can be used to display an error message to the user.

Contacting the Author

I hope you enjoy *AutoCAD Platform Customization* and it changes the way you think about completing your day-to-day work. If you have any feedback or ideas that could improve this book, you can contact me using the following address:

Lee Ambrosius: `lee_ambrosius@hyperpics.com`

On my blog and website you'll find additional articles on customization and samples that I have written over the years. You'll find these resources here:

Beyond the UI: `hyperpics.blogs.com`
HyperPics: `www.hyperpics.com`

If you encounter any problems with this publication, please report them to the publisher. Visit the book's website, `www.sybex.com/go/autocadcustomization`, and click the Errata link to open a form and submit the problem you found.

Chapter 1

Establishing the Foundation for Drawing Standards

Drawing standards, also known as *CAD standards*, are guidelines that help you name the files that are created inside or outside of the Autodesk® AutoCAD® software for a project, the named objects that are used within a drawing file, and the file formats that you might accept. Much like the marketing or management teams use the same logos and memo templates, all drafters or professionals using AutoCAD software in your company should follow a set of company standards.

Without having a well-defined set of standards, you will find it harder to share and output files within a company, and in turn this can lead to delays and make it nearly impossible to achieve a consistent look to all the drawings that your client receives. In addition, it is difficult to customize AutoCAD to help enforce your company's standards if there really are no standards.

Can you imagine what a client might think when a company sends them a set of drawings that contain different fonts or title blocks, or the inconsistent use of lineweights?

Well-established drawing standards ensure that your drawings all look the same when they are presented to the client, and they can make it easier to

- Train new drafters and other professionals on your company's standards that use AutoCAD
- Identify which drawing and externally referenced files are associated with a project
- Determine the purpose of a named object in a drawing
- Share project files with clients and contractors because your standards are well defined

Naming Standards for Projects and Files

As you might have gathered, it is not in your company's best interest to let everyone define their own drafting standards; this same approach applies to naming standards for projects or the files associated with a project, and how files should be stored. At the end of the day, the files created are owned by the company, and there is a pretty good chance that more than one individual will be working on a project over its entire lifetime. There is nothing more frustrating than when changes to a project are requested and the files can't be located because they are missing or no one understands how the files were saved.

The first step your company should consider if you have no current file-naming standard, or if you are considering a change to your existing system, is a way to log and name a project.

Project logging can be as simple as posting a spreadsheet on the network drive or using a project-collaboration site on Microsoft SharePoint to ensure everyone is logging projects using the same system. Once the logging system is determined, you can determine how projects and files should be named. As with the project-logging system, all files should be stored in a central location on a networked drive or a system that allows you to check files in and out, such as Autodesk Vault.

You can take two approaches to the way you name projects and files: you can establish a system yourself using the guidelines that I offer in the next few sections, or you can use the standards set by a consortium or other professional governing body. Based on your industry or the country you work in, you might consider the guidelines established by the American Institute of Architects (`www.aia.org`), National Institute of Building Sciences (`www.nibs.org`), Royal Institute of British Architects (`www.architecture.com`), or American National Standards Institute (`www.ansi.org`).

Project Names

The project-naming structure you choose to use should be short and sequential. For example, you might consider just a basic numeric value such as `000001`, `000002`, and so on. I do not recommend that you use the year as part of the numbering system since projects can span multiple years.

Project names commonly include an alphabetic prefix with one or more letters to make it easy to start a secondary naming system if you decide to organize your projects by business type or some other classification. For example, say your company works on residential, commercial, and government projects. In this case, you might consider prefixing the project's number with `R`, `C`, or `G`, so they would be `R00001`, `C00002`, or `G00003`.

Other information you might want to represent as part of a project name could be

Phase Often a numeric value of one alphabetic letter or number (for example, `-A` or `-1` to represent the first phase of a project).

Location Optional; often a combination of alphabetic letters and numbers to help identify multiple locations on a single and very large job site (for example, `-A1` to represent the first location and `-B1` to represent a second location). Using a Building attribute in a filename might be a better choice for you since all work is being done under a single project instead of multiple projects for a single job.

If a project has only a single phase, it is up to you to determine whether you want to indicate that as part of a project name. A project always has a first phase, but it might never have a second phase, based on the size of the project. Information such as floor, discipline, file type, and sheet type should be reserved for use by files within a project and not included as part of a project's name. Figure 1.1 shows what a structure for a project number might look like.

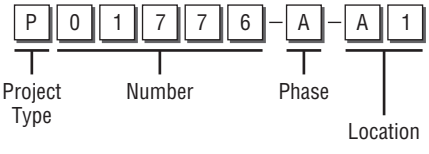


FIGURE 1.1 Possible project numbering structure

The project-naming information I've presented is only one of many possible ways you can define the numbers for your projects. Just keep in mind that project numbers should be short

and sequential. No matter how you choose to name your projects, be sure to document your system and follow it for every project you create. Documenting the system will be important for those inside and outside your company.

After you have defined the requirements and structure for your project name, assign that name to the folder on the network in which all files related to that project should be stored. Alternatively, you can use the name as an attribute in your file-management system of choice.

Filenames

There are two schools of thought when it comes to naming the files that are part of a project: one is to include the project name as part of the filename and the other is not to include it, since it is already the project folder's name. I subscribe to the school of thought that a project name should be part of a file's name, and my main reason is that a file can accidentally be placed in the wrong project folder. If the project name is not part of the filename, the file in a way could be lost forever. No matter which approach you choose, you will want to be consistent. Either prefix all your files with the project name, or don't.

Similar to a project name, the files in a project should use consistent, short, and meaningful names. A basic filename might contain the following information:

Discipline Often a single letter that represents the main discipline that the drawing is used by (for example, `A` for Architecture, `C` for Civil, `M` for Mechanical, or `S` for Structural).

Secondary Discipline Often a single alphabetic letter that helps provide an additional level of classification for the file based on the designated main discipline (for example, `D` for details, `G` for grading, `L` for lighting grid, or `S` for sitework).

Sheet/File Type Often a single alphabetic letter used to identify the contents of a file—for example, `-P` for plans, `-G` for columns grid, and `-I` for images. The same can also be represented by numbers—for example, `-1` for plans, `-0` for columns grid, and `-9` for images. You might want to consider also adding an `-X` when the file should be used only as an external reference across multiple drawing files.

Sheet Number Often a sequential numeric value of two numbers that range from `00` through `99` to uniquely identify a file from other files that might contain all the same file attributes in the project (for example, `01` or `76`).

Figure 1.2 shows a possible structure for a filename based on the file information described.

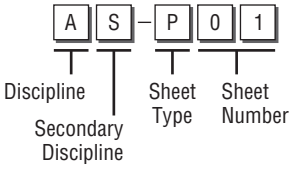


FIGURE 1.2 Possible file-naming structure

The following is some additional information you might want to consider adding to your filenames:

Sheet Size Optional; often a single alphabetic letter used in combination with the sheet number to represent the paper size that the file should be output on (for example, `A` for an ANSI A-size [8 1/2″ × 11″] or `D` for an Architectural D-size [24″ × 36″]).

Building Often a single alphabetic letter used in combination with *floor* to indicate which building the file belongs to (for example, `-A` or `-B`).

Floor Often two numbers to indicate which floor of a building the file belongs to (for example, `-01` or `-22`).

Area Optional; a single alphabetic letter used in combination with *floor* to help identify a specific area on a floor when a floor is broken up into one or more files (for example, `A` or `C`).

Revision Often the letter `R` or `RV` followed by a numeric value to represent the current revision level of a file (for example, `-R1` or `-RV03`). Using revision numbers in a filename has its pros and cons. One of the benefits is that you can go back to an earlier revision of a design if you do not use a system that supports version tracking. The downside is that it affects the other drawings that might reference the drawing; to avoid this, you could create a copy of the drawing being revised and rename the copied file to include the revision number in its name.

Not all of the attributes will make sense for each discipline. For example, a civil drawing will most likely not contain a Building or Floor attribute, but it might contain an Area. So, it is possible that you might use different file-naming structures for different disciplines in your company. If you use different naming structures for different disciplines, be sure the attribute values have the same meaning across the entire company. For instance, do not use numbers to indicate Areas for civil, and alphabetic letters for the architectural drawings.

Managing Standards with Drawing Templates

When you create a new drawing, you have two choices: start from scratch or use a drawing template (DWT) file. Starting from scratch, or using the default drawing that is created when AutoCAD first starts up, is not ideal as it most likely does not conform to your company's standards and the settings can change from release to release. The same is true for the drawing template files that come with AutoCAD: you can use them, but they are not tailored to your company's standards. The default drawing templates make for an excellent starting point, but you should create your own drawing template files so that you know what's in them.

What Is a Drawing Template?

A drawing template is a file that contains the objects, styles, unit of measurement, and other settings that should be used when creating a new drawing with the `new` or `qnew` commands. It has a file extension of `.dwt`. Prior to drawing templates, they were called prototype drawings; you should know that just in case you hear that term come up in a conversation with an AutoCAD veteran.

When a new drawing is created using a DWT file, the DWT file is copied into memory as a new drawing and the DWT file remains unchanged. A DWT file is identical to a drawing (DWG ™) file. While you commonly use the `saveas` command to save a DWG file as a DWT file, you could also just change the file extension of a DWG file from `.dwg` to `.dwt` and achieve the same results.

The following steps explain how to save a DWG file as a DWT file using AutoCAD on Windows:

1. In the AutoCAD software, open the DWG file that you want to save as a DWT file and click the Application button ➢ Save As (or at the command prompt, you could enter **saveas** and press Enter).

2. In the Save Drawing As dialog box, click the Files Of Type drop-down list and choose AutoCAD Drawing Template (*.dwt).

   The AutoCAD software changes the Save In location to the location specified by the Drawing Template File Location node on the Files tab of the Options dialog box (`options` command).

3. In the File Name box, enter a name for the new drawing template file. Click Save.

If you are using AutoCAD on Mac OS, use the following steps:

1. With AutoCAD as the active application, open the DWG file that you want to save as a DWT file and click File ➢ Save As (or at the command prompt, you could enter **saveas** and press Enter).

2. In the Save Drawing As dialog box, click the File Format drop-down list and choose AutoCAD Drawing Template (*.dwt).

   AutoCAD changes the current Save To location in the Where drop-down list to the location specified by the Drawing Template File Location node on the Application tab of the Application Preferences dialog box (`options` command).

3. In the Save As box, enter a name for the new drawing template file. Click Save.

NOTE Drawing standards (DWS) files are always saved in the latest file format. If you support multiple releases of AutoCAD in your company, you will want to instead save your standards to a DWG file with the oldest file format required. Once the DWG file is created, you can then just change the file's extension through the operating system.

Units of Measurement and Format

The drawings that you create in AutoCAD are based on one of two *systems* of measurement: *Imperial* or *English*, and *metric*. Imperial measurement is based on inches, and metric measurement is based on meters. The system of measurement that a drawing currently uses is stored in the `measurement` system variable; 0 (Imperial units) or 1 (metric units). Changing the value of the `measurement` system variable does not affect the objects that are already in a drawing. Resizing objects in a drawing to fit the new system of measurement can be done with the `scale` command.

The measurement system does not affect the formatting of linear and angular units, but it does control the following:

- The default drawing template used for the initial drawing that is created when AutoCAD first starts up: `acad.dwt` for Imperial measurement and `acadiso.dwt` for metric measurement
- Which hatch pattern and linetype definition files AutoCAD looks for in its support-file search paths: `acad.pat` and `acad.lin` for Imperial measurement and `acadiso.pat` and `acadiso.lin` for metric measurement
- The current measurement choice of the Scale List area in the Default Scale List dialog box of the Options dialog box (Windows) and Application Preferences dialog box (Mac OS)

The current measurement system does not affect the way linear and angular drawing units are accepted or displayed. Drawing-unit formatting is controlled by several settings, which you can change by using the Drawing Units dialog box (`units` command); see Figure 1.3.

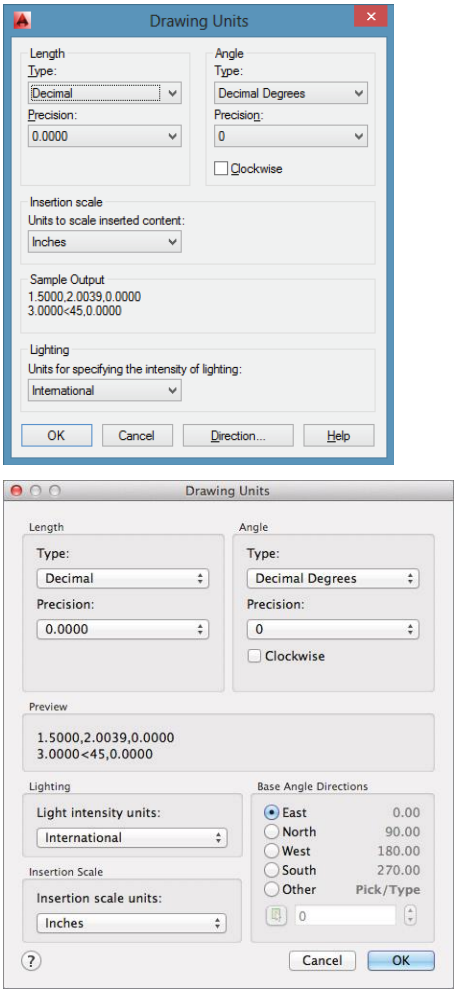


FIGURE 1.3 Drawing Units dialog box. The Windows version of the dialog box appears on the top the Mac OS version appears on the bottom.

Each of the settings shown in the Drawing Units dialog box can also be accessed using system variables. Understanding the relationship between many of the AutoCAD dialog boxes and the system variables they represent will help you automate tasks related to drawing setup, configure drafting aids, and control the default behavior of commands. Table 1.1 explains the system variables that can be changed using the Drawing Units dialog box.

Table 1.1: System variables

Variable	Controls
lunits	Current linear unit mode
luprec	Number of decimal places or the precision in which linear units should be displayed
aunits	Current angular unit mode
auprec	Number of decimal places or the precision with which angular units should be displayed
angdir	Positive direction for angles relative to the active User Coordinate System (UCS)
angbase	Direction of Angle 0 relative to the active UCS
insunits	Units used to properly scale an image or block being attached or inserted into the drawing
lightingunits	Controls the use of generic or photometric lighting along with the lighting units that should be used when rendering the drawing

You can learn more about system variables, including how to change their values, later in this chapter in the section "Working with System Variables." For now, just understand that system variables play an intricate part in being able to customize AutoCAD.

Graphical and Named Objects

As I previously mentioned, a drawing template (DWT) file is the same as a drawing (DWG) file; it can contain both graphical and nongraphical objects. The graphical and nongraphical objects in a DWT file should be limited in number and common to all the new drawings you create. If you have named objects that might be less frequently used, you could store them in a drawing file and then insert that drawing into your drawing to inherit the additional named objects. Or you may choose to automate the creation process of the named objects using one of the available customization methods mentioned later in this book. I cover the creation and management of named objects in Chapter 2, "Working with Named Objects."

Most DWT files have no objects in model space but do have objects placed on one or more of the named layouts in the file. Named layouts commonly have viewports as well as a number of objects or a single block that defines a title block used to frame your design when outputting. The title block could be an external reference (xref) or a block that has already been defined within the DWT file. I cover defining blocks in Chapter 3, "Building the Real World One Block at a Time." General notes, callouts, symbols, and a revision table are other objects that you might place on a layout as well in your DWT file.

The named objects that you should add to your DWT files are as follows:

- Blocks
- Detail view styles
- Dimension styles
- Layers
- Linetypes
- Multileaders
- Multiline styles
- Section view styles
- Table styles
- Text styles

If you work on 3D models, you might also want to consider including UCSs, model-space viewports, saved views, materials, and visual styles. I cover the creation and management of named objects in Chapter 2.

Organize Output with Layouts

Each drawing template (DWT) file contains at least one named layout, and it is a good practice to make sure that the layout is ready to be plotted after a drawing is created using the drawing template. A drawing template can contain more than one layout, but each layout should add some value. If you need additional layouts in your drawing, you could use the `layout` command to duplicate a layout that already exists in the drawing or to import a layout from another drawing file.

The layouts in a drawing template should contain the following:

- Page-setup settings that define the output device, paper size, and other settings that impact the layout when it is plotted or published. I cover configuring plotter devices and plot styles in the section "Defining Plotter Configurations and Plot Styles" later in this chapter.
- A title block that contains all the informational fields needed to identify the drawing once it is plotted or published. A title block might include your company name and address, project name and address, part number, scale, and other information that helps identify the drawing. I discuss creating blocks and attributes in Chapter 3.
- One or more floating viewports that define which areas of model space should be plotted as part of the layout, along with the scale the objects in the viewports should be viewed at.

You might also want to consider adding the following to a drawing for use with or on a layout:

- General notes or disclaimers, and viewport labels that each or most of the drawings created with the drawing template might contain.
- A border around the margins of the paper. The viewports, title block, and other objects on the layout should be inside the border.
- A plot stamp that indicates when the drawing was plotted and the name of the source drawing file.
- Dimension and text styles set up for adding dimensions and annotation to a layout. Remember that layouts are plotted at a 1:1 scale, so the styles need to be defined correctly.

Working with System Variables

System variables in many ways are like the keys hiding under the doormat at the front door of a house. They allow you to directly access many of the settings that can be used to manipulate the AutoCAD environment and control how commands work, instead of you resorting to a dialog box or using options with a command. The values of system variables are stored in one of two places: with the drawing or as part of the user's AutoCAD profile. The user's AutoCAD profile is stored in the Windows Registry, or in several Plist files on Mac OS.

You use the `setvar` command to list and change the current value of a system variable. After you start the `setvar` command, you can enter the name of the system variable to work with or `?` (question mark) to return a list of all values or a filtered list using a wildcard match of the system variables that are supported.

The following steps explain how to list and change the value of a system variable:

1. At the command prompt, enter **`setvar`** and press Enter.
2. At the `Enter variable name or [?]:` prompt, enter the name of the system variable whose value you want to view or change.

   For example, enter **`cursorsize`** and press Enter. The `cursorsize` system variable controls the size of the crosshairs in the drawing window.
3. At the `Enter new value for system_variable_name <current_value>:` prompt, enter a new value for the system variable.

   The current value is displayed between the angle brackets. Pressing Enter without providing a new value ends the command and does not change the value of the system variable.

TIP The name of a system variable can also be entered directly at the command prompt just like any other command. Doing so avoids having to first start the `setvar` command.

You can list all supported variables by doing the following:

1. At the command prompt, enter **`setvar`** and press Enter.
2. At the `Enter variable name or [?]:` prompt, enter **`?`** and press Enter.
3. At the `Enter variable(s) to list <*>:` prompt, press Enter to list all supported system variables or enter a wildcard search string.

   For example, you can enter the wildcard search string **`DIM*`** to list all the system variables that start with the letters `DIM`.

   All the system variables that match the previous entered value are returned.

   If the message `Press ENTER to continue:` is displayed in the command-line window, it indicates that there are additional results that have not yet been displayed. Press Enter to continue paging through all the returned system variables. Continue pressing Enter until you reach the last page of the results returned, or press Esc when you are done paging through the system variables.

Managing Drawing Template Files

After you spend the time creating and customizing your drawing template (DWT) files, you will want to make sure that all users in your company have access to them. By default, AutoCAD looks in a local folder on each workstation for the DWT files it can use. This local folder is specified in the Options dialog box (Windows) or Application Preferences dialog box (Mac OS).

Using a local folder is not ideal for maintaining these files; it is best to change the location AutoCAD looks in to a folder on a networked drive. A networked folder makes it easy to update your drawing template files; post the files once, and then everyone always has access to the latest version.

If you need to use a local folder for your DWT files—when you have remote users, for example—it is good practice to keep your customized files separate from those that come with AutoCAD. Doing so makes it easy to back up your custom files and also removes the temptation to stray from your company's CAD standards. The recommended process for managing local DWT files is to first create a company folder on the local drive, and then create a `Template` subfolder and add your DWT files to the `Template` folder. Keeping the files synchronized from a network or remote server can be done using a batch (BAT) or BASH script (SH), Windows Task Scheduler event, or login script to synchronize the files from a network or remote server.

Follow these steps to specify a different folder for AutoCAD on Windows to look in for DWT files:

1. In Windows Explorer or File Explorer, browse to the folder that you created for this book under `My Documents` (or `Documents`) or create a new folder that you will use to store your DWT files.
   - On the keyboard, press the Windows+E key combination, or right-click in the lower-left corner of the screen (not the AutoCAD application window) and click Windows Explorer or File Explorer, based on your operating system.
   - In Windows Explorer or File Explorer, navigate to the folder where you want to store your DWT files.
   - Right-click in an empty area in the Folders/Files list and click New Folder. Enter the name **`Templates`** or the name you want to use. Press Enter to accept the new name.
2. Click the Application button ➢ Options.
3. In the Options dialog box, click the Files tab.
4. Expand the Templates Settings node, and then expand the Drawing Template File Location node.
5. Double-click the folder path listed under the Drawing Template File Location node.
6. In the Browse For Folder dialog box, browse to and select the folder that contains or will contain your DWT files. Click OK.
7. Click OK to return to your drawing.
8. Click the Application button ➢ New. Click Cancel.

   You should notice that the Select Template dialog box now opens to the location you specified in step 6.

If you are using AutoCAD on Mac OS, use these steps:

1. In Finder, browse to the folder that you created for this book under `Documents` or create a new folder that you will use to store your `DWT` files.
   - In the Mac OS Dock, click Finder or from the desktop click File ➢ Computer.
   - In Finder, navigate to the folder where you want to store your DWT files.
   - Ctrl-click or secondary-click in an empty area in the Folders/Files list and click New Folder. Enter the name **`Templates`** or the name you want to use. Press Enter to accept the new name.
2. Click AutoCAD <*release*> menu ➢ Preferences.
3. In the Application Preferences dialog box, click the Application tab.
4. Expand the Templates Settings node, and then expand the Drawing Template File Location node.
5. Double-click the folder path listed under the Drawing Template File Location node.
6. In the Open dialog box, browse to and select the folder that contains or will contain your DWT files. Click Open.
7. Click OK to return to your drawing.
8. Click File ➢ New Drawing. Click Cancel.

   You should notice that the Select Template dialog box now opens to the location you specified in step 6.

In addition to specifying the location of the Select Template dialog box, you can specify the default DWT file that is used with the `qnew` (Quick New) command. You specify the DWT file for the `qnew` command using the Options dialog box (Windows) or Application Preferences dialog box (Mac OS).

Use these steps to specify which DWT file should be used when the `qnew` command is executed in AutoCAD on Windows:

1. Click the Application button ➢ Options.
2. In the Options dialog box, click the Files tab.
3. Expand the Templates Settings node, and then expand the Default Template File Name for QNEW node.
4. Double-click the filename listed under the Default Template File Name For QNEW node.
5. In the Select A File dialog box, browse to and select the DWT file that you want to use with the `qnew` command. Click Open.
6. Click OK to return to your drawing.
7. On the Quick Access toolbar, click New. A new drawing file is created based on the DWT file you selected in step 5.

If you are using AutoCAD on Mac OS, use the following steps:

1. Click AutoCAD <*release*> menu ➢ Preferences.
2. In the Application Preferences dialog box, click the Application tab.
3. Expand the Templates Settings node, and then expand the Default Template File Name For QNEW node.
4. Double-click the filename listed under the Default Template File Name For QNEW node.
5. In the Open dialog box, browse to and select the DWT file that you want to use with the `qnew` command. Click Open.
6. Click OK to return to your drawing.
7. At the command prompt, enter **`qnew`** and press Enter. A new drawing file is created based on the DWT file you selected in step 5.

Choosing a File Format for Your Drawings

Out of the box, AutoCAD saves all the drawings you create or open to the latest file format. This is done to ensure that any of the objects you create are properly stored and can be restored when the drawing is opened later. Using the current drawing file format is not always the best choice; perhaps you are working with another department that needs access to the drawings you create and they are still on an older release, or the contract you are bidding on requires all the files for a project to be submitted in a specific format.

No matter the reasoning, AutoCAD allows you to set the default file format it uses when saving a drawing instead of requiring you to change to an earlier format from the Files Of Type (Windows) or File Format (Mac OS) drop-down list in the Save Drawing As dialog box. The default file format for a drawing can be set in the Options dialog box (Windows) or Application Preferences dialog box (Mac OS).

Follow these steps to change the default file format used when saving a drawing file in AutoCAD on Windows:

1. Click the Application button ➢ Options.
2. In the Options dialog box, click the Open And Save tab. In the File Save area, click the Save As drop-down list and then choose the default file format you want to use.
3. Click OK.

If you are using AutoCAD on Mac OS, use the following steps:

1. Click AutoCAD <*release*> menu ➢ Preferences.
2. In the Application Preferences dialog box, click the General tab. Then choose the default file format you want to use from the Save As drop-down list in the File Save area.
3. Click OK.

TIP Before you save a drawing file to an earlier file format, you will want to know which releases your users are working in. I don't recommend just saving all your drawing files to the oldest possible release, as that could result in some undesired results. Some objects in a drawing could disappear or be exploded into individual objects.

Table 1.2 shows which file format you should use to save your drawings so that users on an older release can open your files.

Table 1.2: AutoCAD releases and drawing file formats

AutoCAD Release(s)	Drawing File Format
AutoCAD 2013 and AutoCAD 2014	AutoCAD 2013
AutoCAD 2010, AutoCAD 2011, and AutoCAD 2012	AutoCAD 2010
AutoCAD 2007, AutoCAD 2008, and AutoCAD 2009	AutoCAD 2007
AutoCAD 2004, AutoCAD 2005, and AutoCAD 2006	AutoCAD 2004
AutoCAD 2000, AutoCAD 2000i, and AutoCAD 2002	AutoCAD 2000
AutoCAD Release 14	AutoCAD R14
AutoCAD Release 13	AutoCAD R12 DXF
AutoCAD Release 12	AutoCAD R12 DXF

If you want to make sure that all the files in a project are saved in a specific file format, you can use one of the following tools:

Autodesk DWG TrueView A drawing file conversion program that is standalone from AutoCAD. It's available on Windows only and can be downloaded from `www.autodesk.com/dwg`.

AutoCAD 360 Online collaboration website that allows you to view, edit, and share DWG files with others on a project. After a drawing has been uploaded, it can be downloaded in a different file format. The AutoCAD 360 website can be found at `https://www.autocadws.com/`.

`ETRANSMIT` command Command that can be used to resave all the drawing files in a sheet set, or in the drawing files you add to the Create Transmittal dialog box. It's available on Windows only.

Script Pro Utility that allows you to run script files on selected drawing files. The utility is available for Windows only and can be downloaded from here:

`http://usa.autodesk.com/adsk/servlet/item?siteID=123112&id=4091678&linkID=9240618`

Script Files and `SCRIPT` command You can create a script file that opens, saves, and closes multiple drawing files. I cover creating script files in Chapter 8, "Automating Repetitive Tasks."

Defining Plotter Configurations and Plot Styles

Plotting, printing, and publishing are all forms of outputting a drawing file to a hardcopy (a physical sheet of paper) or an electronic representation of a hardcopy. Doing so helps you keep

your intellectual property in your drawings secure, and it also gives those that do not have access to or know how to use AutoCAD the ability to review and approve your drawings.

Before you can output a drawing, you need to

- Set up an output device
- Define the plot styles that control the way drawing objects appear in the output
- Configure the settings of model space or a named layout

Configuring Output Devices

Setting up an output device is commonly handled by installing a system printer in Windows or Mac OS, but AutoCAD on Windows also supports a second option that allows you to configure a nonsystem printer using custom device drivers. Nonsystem printers are stored in PC3 files. Once a device is set up, additional settings can be specified inside AutoCAD to control the output being sent to the device. After a device is configured, you can then assign the device directly to a layout or page setup, or set it as the current device when using the `plot` command.

You can configure a nonsystem printer for use with AutoCAD on Windows by doing the following:

1. Click the Application button ➢ Print ➢ Manage Plotters.
2. In the Plotters window, double-click Add-A-Plotter Wizard.
3. In the Add Plotter wizard, click Next on the Introduction page.
4. On the Begin page, select one of the three types of plotters that can be added:
   - My Computer: The device is configured to use a port on your local computer, to plot to a file, or to use an AutoSpool utility (an application that controls how the plot file is handled).
   - Network Plotter Server: The device is configured by posting plot files to a network location where the plotter checks for new files that need to be plotted.
   - System Printer: The device is configured to use an installed system printer and allows you to control the properties of AutoCAD-specific output settings.

   Then click Next.
5. You will see a different page based on which option you selected:
   - If you chose My Computer, the Plotter Model page is displayed. Select a manufacturer and one of the supported plotter models. Click Next.
   - If you chose Network Plotter Server, the Network Plotter page is displayed. Enter or select the name of the network server (UNC) that you want to use, and click Next. The Plotter Model page is displayed. Select a manufacturer and one of the supported plotter models. Click Next.
   - If you chose System Printer, the System Printer page is displayed. Select a printer that is installed under the operating system and click Next.
6. If the Import PCP Or PC2 page is displayed, click Import if you have a PCP or PC2 file from AutoCAD Release 14 or earlier that contains plotter-specific information. Select the PCP or PC2 file and click Open. Click Next to move to the next page in the wizard.

7. If you chose System Printer in step 4, the Ports page is displayed. Select a port on your local computer, plot to a file, or use an AutoSpool utility. Click Next.
8. On the Plotter Name page, enter a name in the Plotter name text box and click Next.
9. On the Finish page, click Edit Plotter Configuration to make changes to the device's settings in the Plotter Configuration Editor (see Figure 1.4) or click Calibrate Plotter to adjust the output size compared to that of a test drawing in the Calibrate Plotter wizard. Both of these options are optional, and availability is based on the type of device you are adding along with the plotter model you specified. Click Finish.

   The new plotter is added to the Plotter window and is ready for use by AutoCAD.

Do the following to edit a nonsystem printer in AutoCAD on Windows:

1. Click the Application button ➤ Print ➤ Manage Plotters.
2. In the Plotters window, double-click the PC3 file that contains the nonsystem printer configuration you want to edit. The Plotter Configuration Editor (see Figure 1.4) is displayed.

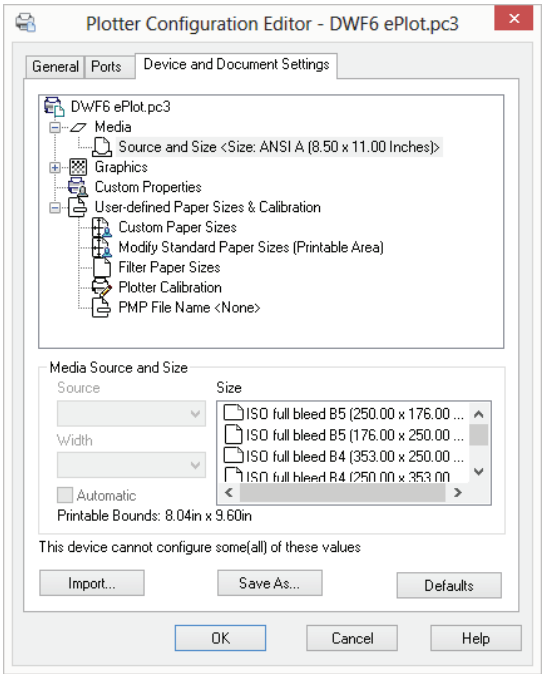


FIGURE 1.4
Editing a plotter configuration file

3. Optionally, in the Plotter Configuration Editor on the General tab, enter a description for the plotter configuration in the Description text box.
4. Click the Ports tab and change the current port, plot to file, or AutoSpool utility settings.
5. Click the Device And Document Settings tab and change device settings as needed. Click OK.

After a system printer is installed or a plotter configuration is created, you can use it to output the objects in model space or on a named layout. The following steps explain how to assign a plot configuration to a layout or page setup and how to use it with the `plot` command:

- For a layout or page setup, click the Application button ➢ Print ➢ Page Setup (Windows) or click File ➢ Page Setup Manager (Mac OS). In the Page Setup Manager, select the layout or page setup you want to change, and then click Modify on Windows or click the Action menu (gear icon) ➢ Edit on Mac OS to display the Page Setup dialog box. In the Page Setup dialog box, in the Printer/Plotter area, choose a device from the Name (Windows) or Printer (Mac OS) drop-down list. Click OK. Click Close to exit the Page Setup Manager.
- For plotting or printing with the `plot` command, click the Application button ➢ Print (Windows) or click File ➢ Print (Mac OS). In the Plot dialog box (Windows) or Print dialog box (Mac OS) in the Printer/Plotter area, choose a device from the Name (Windows) or Printer (Mac OS) drop-down list. Click OK (Windows) or Print (Mac OS) to output the drawing.

Using and Creating Plot Styles

Plot styles are used to control the way drawing objects appear onscreen and when they are output to hardcopy (a physical sheet of paper) or an electronic representation of a hardcopy. AutoCAD supports two types of plot styles: *color dependent* and *named*.

Color-dependent plot styles, stored in CTB files, are the most common of the two styles that AutoCAD supports. They are the way object properties were applied during plotting in AutoCAD prior to the introduction of named plot styles in AutoCAD 2000. With color-dependent plot styles, as the name reflects, the color assigned to an object in a drawing controls the object properties that are assigned during plotting. Even though your drawings can use true colors, plot styles are based on the AutoCAD Color Index (ACI) system of 255 colors, and true color values are therefore mapped to their nearest ACI value when plotting.

Named plot styles, stored in STB files, are the newest way to control object properties when plotting. First introduced in AutoCAD 2000, this style uses a name-based system instead of being dependent on the color of an object. Although this might sound ideal, there are a few places where color can be assigned only to objects in a drawing or style. For example, you can only set the color of the grid lines in a table or the dimension line of a dimension style.

TIP You can convert a drawing from the color-dependent plot style to the named plot style, and vice versa, by using the `convertpstyles` command.

Both types of plot styles have their advantages. Color dependent is based on the legacy system and is thus compatible with decades of drawings, making it the clear choice for most companies. It is also the style that provides the most control over complex objects and many styles. Switching to named plot styles requires a bit of planning and time, which at the end of the day keeps it from being an option for many companies. Here are a few of the issues you need to consider if you plan on adopting named styles:

- Existing block libraries need to be updated to ensure they display correctly using named plot styles.
- Dimension and table styles might need to be updated to ensure they use a single color and that they use lineweights and linetypes to control the way they output.

- Color-dependent drawings need to be updated to use named plot styles with the `convertpstyles` command. Having a mix of plot styles in the same project does not give your drawings a consistent look when they are plotted.

You can create a plot style on Windows by doing the following:

1. Click the Application button ➢ Print ➢ Manage Plot Styles.
2. In the Plot Styles window, double-click Add-A-Plot Style Table Wizard.
3. In the Add Plot Style Table wizard, click Next on the Introduction page.
4. On the Begin page, select one of these four options to create a plot style:
   - Start From Scratch: Use this option when you want to create a new color-dependent (CTB) or named (STB) plot style file.
   - Use An Existing Plot Style Table: This option creates a copy of an existing CTB or STB file, and then allows you to edit the copied file.
   - Use My R14 Plotter Configuration (CFG): If upgrading from AutoCAD Release 14, you can import your settings to create a new plot style file.
   - Use A PCP Or PC2 File: If you have a PCP or PC2 file that was exported from an earlier release, you can import the pen table properties and create a new plot style file.

   Then click Next.
5. If the Table Type page is displayed, click Color-Dependent Plot Style Table or Named Plot Style Table. Click Next.

   If the Browse File Name page is displayed, browse to and select the type of file that is expected and any additional information that is needed for the type of file you selected. Click Next.
6. On the File Name page, enter a name in the File Name text box for the new plot style file. Click Next.
7. On the Finish page, click Plot Style Table Editor to make changes to the new plot style table in the Plot Style Table Editor (see Figure 1.5). Click Finish.

If you are using AutoCAD on Mac OS, use the following steps to create a new plot style file:

1. Click File ➢ Plot Styles.
2. In the Plot Styles window, double-click one of the plot styles list. The Plot Style Table Editor (see Figure 1.5) is displayed.
3. From the AutoCAD Plot Style Editor menu bar, click File ➢ New Color-Based Plot Style Table (CTB) or New Named Plot Style Table (STB).
4. In the Plot Style Table Editor, click Save As.
5. In the Save dialog box, enter a name for the new file in the Save As text box and specify a location to save the file to. Click Save.

6. In the Plot Style Table Editor, make the changes to the plot styles properties and click Save & Close.

If you want to make changes to an existing plot style file, do the following:

1. Click the Application button ➤ Print ➤ Manage Plot Styles (Windows) or click File menu ➤ Plot Styles (Mac OS).
2. In the Plot Styles window, double-click the plot style you want to edit. The Plot Style Table Editor (see Figure 1.5) is displayed.

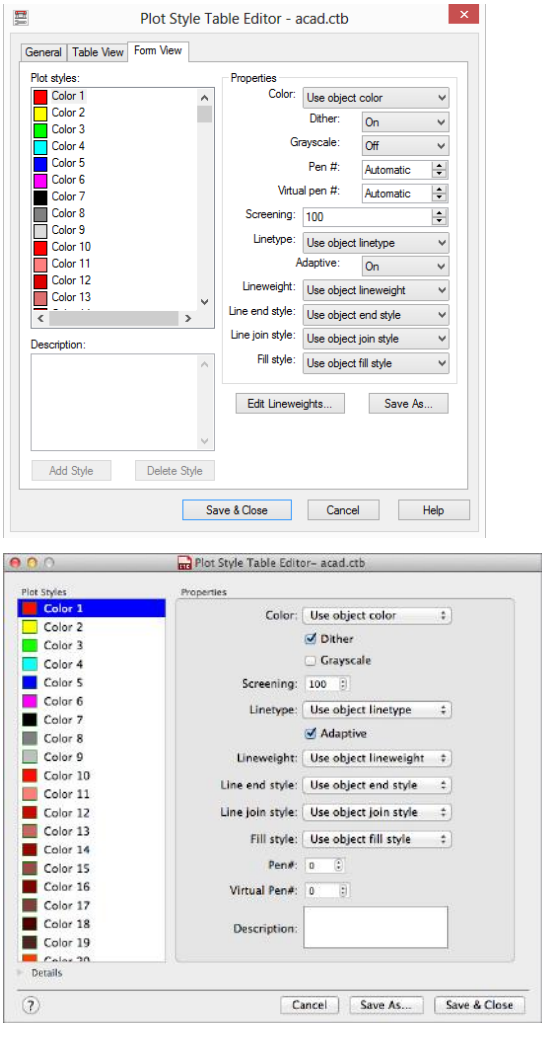


FIGURE 1.5 Editing a plot style file. The Windows version of the dialog box appears on the top and the Mac OS version on the bottom.

3. In the Plot Style Table Editor, edit the properties of the plot style as needed. If you are working with a named plot style, create any new named styles that you need. In Windows, click the Form View tab; it is the easiest way to work with the plot styles in the file.

4. Click Save & Close to save the changes you made.

After a plot style has been created, you can use it control how the objects of your drawing appear when output. The following steps explain how to assign a plot style to a layout or page setup, and how to use it with the `plot` command:

- For a layout or page setup, click the Application button ➢ Print ➢ Page Setup (Windows) or click File ➢ Page Setup Manager (Mac OS). In the Page Setup Manager, select the layout or page setup you want to change, and then click Modify on Windows or click the Action menu (gear icon) ➢ Edit on Mac OS to display the Page Setup dialog box. In the Page Setup dialog box, in the Plot Style Table area, choose a plot style from the drop-down list. If prompted to apply the plot style to all layouts, click Yes. On Mac OS only, you might need to click the More Options button in the lower-right corner of the dialog box to see the Plot Style Table area. Click OK. Click Close to exit the Page Setup Manager.
- On Windows, for plotting with the `plot` command, click the Application button ➢ Print. In the Plot dialog box, click the More Options button in the lower-right corner of the dialog box. In the Plot Style Table area, choose a plot style from the drop-down list. Click OK to output the drawing.
- On Mac OS, for printing with the `plot` command, click File ➢ Print. In the Print dialog box, in the Page Setup area, click Edit Page Setup. In the Page Setup dialog box, in the Plot Style Table area, choose a plot style from the drop-down list. You might need to click the More Options button in the lower-right corner of the dialog box to see the Plot Style Table area. Click Print to output the drawing.

Managing Plotter Configuration and Plot Style Files

After you have added your output devices and created your plot styles, you will want to make sure that all users in your company have access to them. By default, AutoCAD looks in local folders on each workstation for the PC3, CTB, and STB files that it might need to output your drawing files. These local folders are specified in the Options dialog box (Windows) or Application Preferences dialog box (Mac OS). As with other custom files, such as DWT files, you should place all your common plotter configuration and plot style files on a network location.

Follow these steps to specify different folders for AutoCAD on Windows to look in for PC3 and CTB/STB files:

1. Click the Application button ➢ Options.
2. In the Options dialog box, click the Files tab.
3. Expand the Printer Support File Path node, and then expand the Print Configuration Search Path node to change the location for PC3 files or expand the Plot Style Table Search Path node to change the location for CTB/STB files.
4. Double-click the folder path listed under the expanded node.
5. In the Browse For Folder dialog box, browse to and select the folder that contains your PC3 or CTB/STB files. Click OK.

   You can also select the Print Configuration Search Path or Plot Style Table Search Path node and click the Add button to add an additional location so you have access to your custom files and those that come with AutoCAD.

6. Click OK to return to your drawing.

If you are using AutoCAD on Mac OS, use the following steps:

1. Click AutoCAD <*release*> menu ➢ Preferences.
2. In the Application Preferences dialog box, click the Application tab.
3. Expand the Printer Support File Path node, and then expand the Print Configuration Search Path node to change the location for PC3 files or expand the Plot Style Table Search Path node to change the location for CTB/STB files.
4. Double-click the folder path listed under the expanded node.
5. In the Open dialog box, browse to and select the folder that contains your PC3 or CTB/STB files. Click Open.

   You can also select the Print Configuration Search Path or Plot Style Table Search Path node and click the + (plus) button to add an additional location so you have access to your custom files and those that come with AutoCAD.
6. Click OK to return to your drawing.

Enforcing CAD Standards

CAD standards enforcement is a gradual process if you are not doing it today. The best strategy you have is positive reinforcement and coaching. At the end of the day, your drafters and professionals need to want to follow all the established standards. It can be tempting to take the quick and easy route to get a job done, but shortcutting a process can have unexpected consequences, such as a custom tool not working in a drawing and resulting in manual steps or a drawing not plotting correctly. After all, a process works each and every time only when each step is completed properly.

There are steps you can take to help make following CAD standards easier for the drafters and professionals you support. You can use these methods to apply and enforce your company's standards:

User-Interface Customization AutoCAD offers an interface that can be customized to the way your company works. You can create ribbon buttons or menu items that use a custom macro to set a layer or style as current before starting a command. Customizing the user interface is covered in Chapter 5, "Customizing the AutoCAD User Interface for Windows," and Chapter 6, "Customizing the AutoCAD User Interface for Mac."

Scripts Script files are a great way to execute multiple commands and options in a specific order using predefined values without needing to understand a programming language. You can create layers and styles, insert a title block on a specific layer, or even change your current drafting settings. Creating script files is discussed in Chapter 8.

Action Macros Created using the Action Recorder, action macros are a modern take on script files. They allow you to execute multiple commands and options in a specific order, but they are created interactively while you use AutoCAD. After an action macro is created, it can

be executed in other drawing files. Recording action macros is covered in Chapter 8. (Action macros are not available on AutoCAD for Mac OS.)

Custom Programs The programming languages that AutoCAD supports give you the most flexibility in enforcing CAD standards. Custom programs can be created to make sure a specific layer is current before a command is started, whether the user starts the command from the user interface or from the command prompt.

You can also use custom programs to step through the objects in a drawing and make sure they conform to your standards, and much more.

Tool Palettes An arrangement of tools in the Tool Palettes window allows you to create objects using specific property settings. You can create a Note tool that uses a specific combination of text style and layer for general notes and define another tool for disclaimers; access and insert commonly used blocks at a set scale; or fill an enclosed area with a hatch pattern, using a specific lineweight and color. I discuss tool palettes in Chapter 7, "Creating Tools and Tool Palettes." (The Tool Palettes window is not available on AutoCAD for Mac OS.)

DesignCenter™ DesignCenter allows you to access the named objects defined in a drawing and add them to your current drawing. It also allows you to insert blocks as well as attach external drawings and raster images. I do not cover using DesignCenter in this book; use the AutoCAD Help to learn more about this feature. (DesignCenter is not available on AutoCAD for Mac OS.)

Content Explorer™ Content Explorer is a modernized version of DesignCenter that allows you to locate named objects from other drawing files on a network and add them to your current drawing. I do not cover configuring and using Content Explorer in this book; use the AutoCAD Help to learn more about this feature. (Content Explorer is not available on AutoCAD for Mac OS.)

AutoCAD on Windows offers multiple tools that are designed to help you validate named objects and their properties in a drawing against those defined in a drawing standards (DWS) file. These tools are as follows:

Standards Manager Used to configure and validate a drawing file for standards violations. Violations found can be fixed to conform to the standards you are validating against.

Batch Standards Checker Validates the standards in a set of drawing files and generates a report of all the violations found and which drawing files they were found in. You must open each drawing file manually in order to fix the violations found.

Layer Translator Used to create layer mappings and align the layers of a drawing with those defined in your standards.

Drawing Standards (DWS) Files (Windows Only)

A drawing standards (DWS) file is the same as a drawing or a drawing template file, with the exception of the content it contains and its file extension. A DWS file commonly does not contain any graphical objects, but it does contain all of the named objects that are used to define your CAD standards and your layer translation mappings for the Layer Translator. For more

information on named objects, see Chapter 2. I cover the Layer Translator in the "Translating Layers (Windows Only)" section later in this chapter.

At a minimum, your DWS files should contain the following named objects:

- Dimension styles
- Layers
- Linetypes
- Text styles

These steps explain how to save a DWG file to a DWS file using AutoCAD on Windows:

1. In AutoCAD, open or create a DWG file that contains the named objects that represent your CAD standards. Click the Application button ➢ Save As.
2. In the Save Drawing As dialog box, click the Files Of Type drop down-list and choose AutoCAD Drawing Standards (*.dws).
3. Browse to a common location on the network to store the DWS file so that you and others in your company can access it.
4. In the File Name box, enter a name for the new drawing template file. Click Save.

Drawing standards (DWS) files are always saved in the latest file format. If you support multiple releases of AutoCAD in your company, you will want to instead save your standards to a DWG file with the oldest file format required. Once the DWG file is created, you can then just change the file's extension through the operating system.

Configuring, Checking, and Fixing Drawing Standards (Windows Only)

The AutoCAD Standards Manager provides you with the ability to check several of the named object types in a drawing against those defined in a drawing standards (DWS) file. Using the Standards Manager requires the completion of two distinct processes: first, you associate one or more DWS files that contain your CAD standards with the current drawing, and second, you check and fix any standards violations found.

Configuring Standards

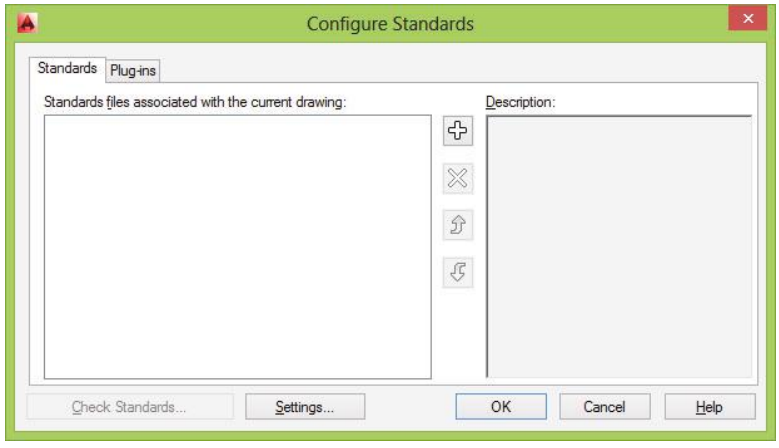
Configuring a drawing to be checked for standards violations requires you to associate the DWS files that contain your CAD standards and specify which plug-ins you want to use. A plug-in defines the comparison rules that should be used to find any standards violations in the current drawing against the approved CAD standards in the DWS files. You can choose from one of four plug-ins that come with AutoCAD, you can obtain plug-ins from third-party developers, or you can develop your own plug-ins using the CAD Standards Plug-in API. You associate DWS files with a drawing using the `standards` command, which displays the Configure Standards dialog box.

TIP You can associate DWS files with a drawing template (DWT) file. Any new drawings created from the drawing template will maintain the associations so that you do not have to go through the configuration process for each new drawing file you create.

The following steps explain how to associate a DWS file with a drawing and specify which standards plug-ins to use:

1. On the ribbon, click Manage tab ➢ CAD Standards panel ➢ Configure (or at the command prompt, enter **`standards`** and press Enter). The Configure Standards dialog box (see Figure 1.6) is displayed.

FIGURE 1.6
Configuring drawing standards files



2. In the Configure Standards dialog box, click the Standards tab.

   You use the Standards tab to associate DWS files with the current drawing and to control the order in which AutoCAD should search the associated DWS files when validating standards. AutoCAD starts searching for named objects that match the names of those in the current drawing from the top of the list downward.

3. Click the + (plus) button to add a new DWS file association to the current drawing. The Select Standards File dialog box is displayed.

4. In the Select Standards File dialog box, browse to and select the DWS file that you want to associate with the drawing. Click Open.

   Only one DWS file can be associated at a time. Click the + (plus) button and add all the DWS files you want to check against.

5. If you add more than one DWS file, select an associated DWS file and click Move Up /Move Down to change the search order of the DWS files.

   Select a DWS file and click the X to remove an associated DWS file.

6. Click the Plug-ins tab. You use the Plug-ins tab (see Figure 1.7) to specify which plug-ins should be used when checking the standards of the current drawing against the DWS files that have been associated with the drawing.

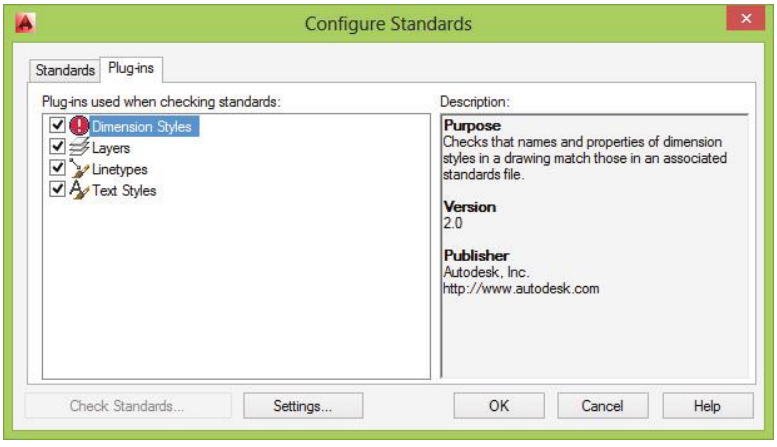


FIGURE 1.7
Enabling the plug-ins to use when validating drawing standards

7. Specify the plug-ins you want to use when checking for standards violations.
8. Click the Settings button. The CAD Standards Settings dialog box (see Figure 1.8) is displayed.

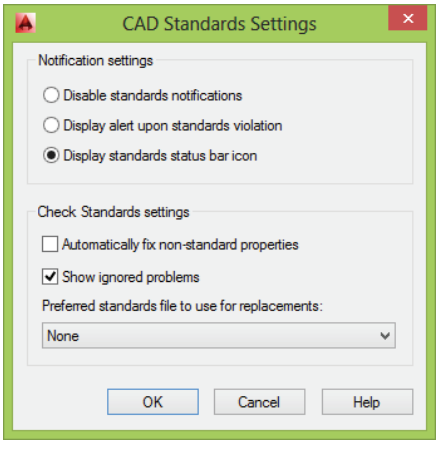


FIGURE 1.8
Changing the notification and Check Standards settings

9. In the CAD Standards Settings dialog box, change the settings as needed and click OK.

   You can change the display behavior of the CAD Standards notification balloon and icon in the drawing status bar tray (see Figure 1.9) in the Notification Settings section. The settings in the Check Standards Settings section control how errors are handled when a drawing is checked for standards violations.

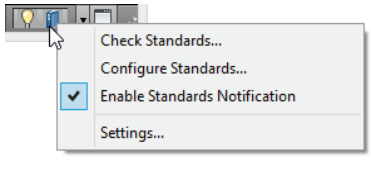


FIGURE 1.9
Access to CAD Standards notifications and settings is just a click away.

10. Click OK to close the Configure Standards dialog box and return to your drawing.

    Based on your settings in the CAD Standards Settings dialog box, the CAD Standards icon might be displayed in the drawing status-bar tray. The tray icon, when displayed, can be used to indicate whether at least one DWS file is associated with the current drawing. Right-clicking the icon allows you to configure and check the drawing standards in your drawing.

Checking For and Fixing Standards Violations

After you have associated at least one DWS file to a drawing and specified which plug-ins to use, you can check your drawing for standards violations. While you are checking for standards violations, each plug-in is executed one by one, and how a plug-in checks for standards violations can vary between plug-ins.

The standard four plug-ins that come with AutoCAD compare the name of an object in the current drawing to the names of objects in the associated DWS files. When a match is found, AutoCAD checks to see whether the properties of the two named objects are the same. If they are the same, AutoCAD moves on to the next named object and checks it. If the properties are different, you are asked to fix or ignore the standards violation. If no matching named object is found between the drawing and DWS files, you are prompted to select one of the approved named objects from the associated DWS files or ignore the standards violation.

Follow these steps to check a drawing for any standards violations:

1. On the ribbon, click Manage tab ➢ CAD Standards panel ➢ Check (or at the command prompt, enter **`checkstandards`** and press Enter).

   The Check Standards dialog box (see Figure 1.10) is displayed. If no DWS file is associated with the drawing file, you will be presented with an error message. Click OK to configure the CAD standards settings for the drawing and then continue checking for standards violations.

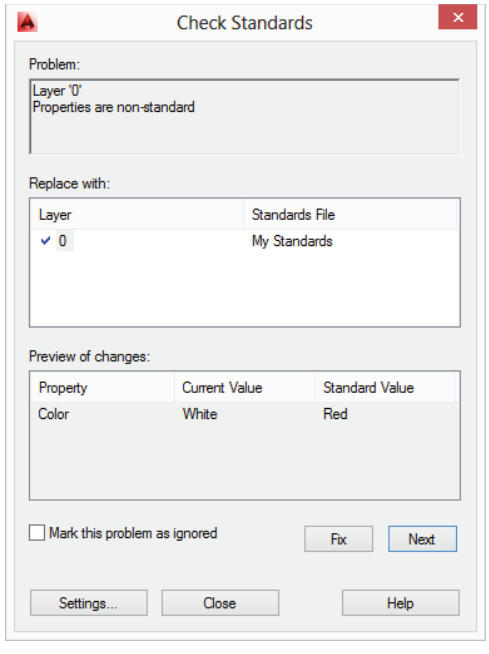


Figure 1.10
Standards violation found

2. In the Check Standards dialog box, in the Problem area review the first standards violation found. The Problem area provides you with a description of the standards violation.
3. In the Replace With area, choose one of the available fixes.

   The Replace With area provides you with all the possible fixes that one of the plug-ins identified for the standards violation. A plug-in can recommend a fix that is indicated by a blue check mark to the left of the item. After selecting a possible fix for the standards violation, you can see an overview of the changes that will be made in the Preview Of Changes area. You can have AutoCAD automatically fix standards violations that have nonstandard properties by clicking the Settings button and enabling Automatically Fix Non-standard Properties in the CAD Standards Settings dialog box. If your DWS files offer more than one recommendation for a fix, you can specify the preferred DWS file that you want to use from the CAD Standards Settings dialog box as well.
4. Click Fix to correct or Next to skip the standards violation, and move to the next one found.

   When you skip a standards violation, you can click Mark This Problem As Ignored to not have the violation show up the next time the drawing is checked. You can choose to display all ignored violations by clicking the Settings button and enabling Show Ignored Problems in the CAD Standards Settings dialog box.
5. Continue stepping through each standards violation until the Check Standards - Check Complete message box is displayed. Click Close.

   The Check Standards - Check Complete message box (see Figure 1.11) provides you with a summary of the actions taken to resolve the standards violations found.

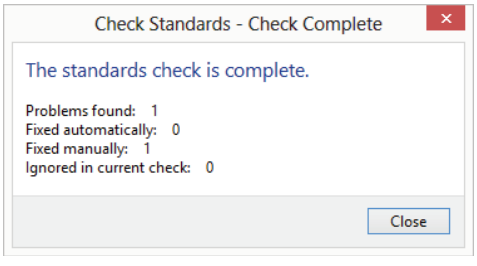


FIGURE 1.11 The check for standards violations is complete.

6. Click Close to close the Check Standards dialog box and return to your drawing.

While working on a drawing with CAD standards configured, you might see the Standards Violation balloon (see Figure 1.12) come up. This indicates that something in the drawing conflicts with your company's standards. A custom routine might have caused the standards violation, or maybe you inserted a drawing that contains named objects that were not updated with the latest standards. Click the Run Check Standards link in the balloon to fix or ignore the standards violation.

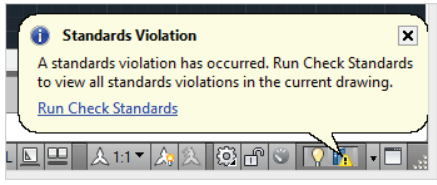


FIGURE 1.12
Standards violation detected

CHECKING DRAWINGS IN BATCHES

The Check Standards dialog box is efficient for checking the standards of the current drawing, but it is not ideal if you have five, ten, or even hundreds of drawings in your project that need to be checked. When AutoCAD is installed, it also installs an external utility called the Batch Standards Checker. The Batch Standards Checker allows you to select a number of drawings and check them against the standards defined in a DWS file.

The Batch Standards Checker allows you to check the drawing files using the DWS files that are already associated with each file, or you can specify which DWS files should be used. This utility uses the same plug-ins that are available in the Configure Standards dialog box. The one feature that the Batch Standards Checker does not support is the ability to fix any of the violations found; that must be done by opening each drawing and using the `checkstandards` command.

The following steps explain how to use the Batch Standards Checker:

1. Do one of the following:
   - (Windows XP and Windows 7) Click the Windows Start button ➤ [All] Programs ➤ Autodesk ➤ AutoCAD *<release>* ➤ Batch Standards Checker.
   - (Windows 8) On the Start screen, right-click and click All Apps. Under the AutoCAD *<release>* category, click Batch Standards Checker.

   The Batch Standards Checker dialog box (see Figure 1.13) is displayed.
2. In the Batch Standards Checker, on the Drawings tab click the + (plus) button.
3. In the Batch Standards Checker - File Open dialog, browse to and select the drawings and drawing templates you want to check for standards violations. Click Open.

   Press and hold the Ctrl key when browsing to select more than one file.
4. Optionally, click the Check External References Of Listed Drawings check box.

   When this option is enabled, the Batch Standards Checker will also check the external references attached to the listed files for any standards violations.
5. Click the Standards tab.

   This tab is similar to the Standards tab of the Configure Standards dialog box in AutoCAD. The only difference is the two options located along the top of the tab.

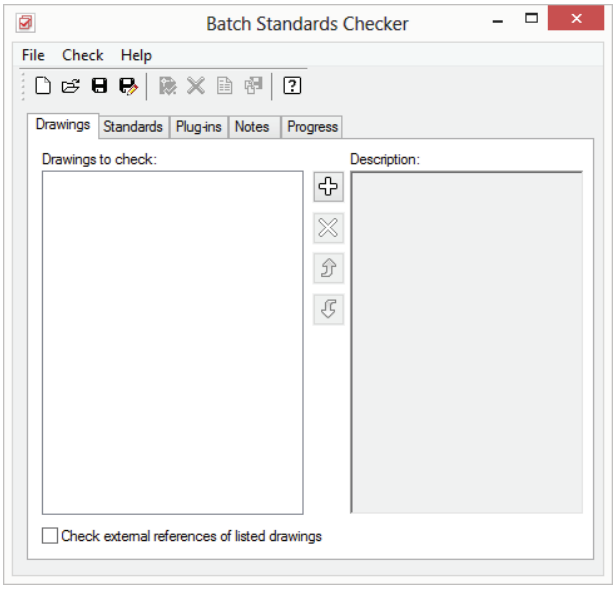


FIGURE 1.13
More than one drawing can be checked for standards violations with the Batch Standards Checker.

6. Choose one of the two options along the top of the tab:
   - The Check Each Drawing Using The Associated Standards Files option instructs the Batch Standards Checker to use the DWS files associated with each drawing file; if no DWS file is associated with a drawing file, the drawing is not checked for standards violations.
   - Use the Check All Drawings Using The Following Standards Files option to specify which DWS files to use for validating the CAD standards. Click the + (plus) button to list which DWS files to use. If more than one DWS file is added, select an associated DWS file and click Move Up/Move Down to change the search order of the DWS files. Select a DWS file and click the X to remove an associated DWS file.
7. Click the Plug-ins tab.

   This tab is identical to the Plug-ins tab of the Configure Standards dialog box in AutoCAD.
8. Specify which plug-ins to use when checking the drawings for standards violations.
9. Optionally, click the Notes tab and enter the text for the note.

   You might want to list the project's name, your name, and any other information that could be useful to the next person who views the report generated after checking the drawing files for standards violations.
10. On the Batch Standards Checker toolbar, click Save.
11. In the Batch Standards Checker - File Save dialog, browse to a location and enter a name for the Standards Check (CHX) file. Click Save.

The CHX file contains a reference to the drawings you added to the list and specifies which drawing standards (DWS) files and plug-ins to use, along with the notes and other settings you included. You might want to create a CHX file for each one of your projects and then store them with the drawing files in your project.

12. On the Batch Standards Checker toolbar, click Start Check.

The Progress tab is displayed with the current progress of the standards violation check for the drawing files that were listed on the Drawings tab. Once processing has been completed, the Standards Audit Report (see Figure 1.14) is displayed in your default web browser. Use this report to view any standards violations found. The most recent report is saved with the CHX and can be viewed by clicking Check ➢ View Report on the Batch Standards Checker's menu bar.

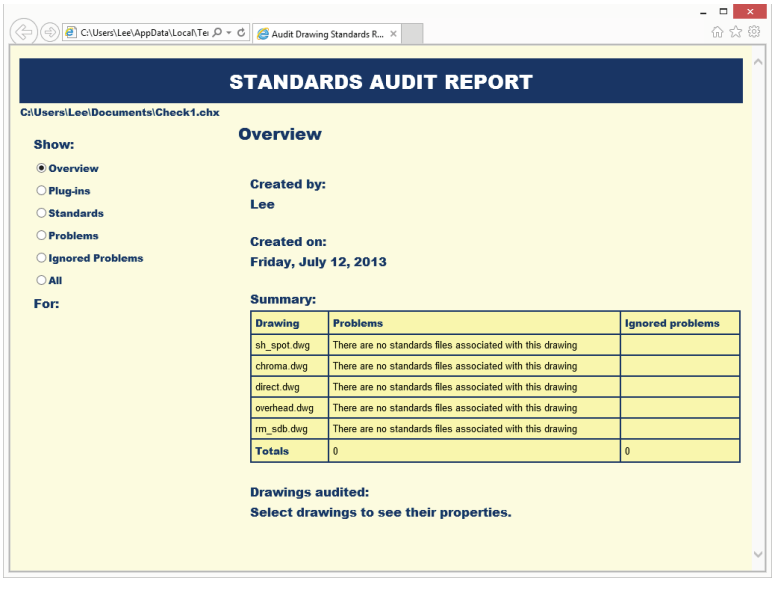


FIGURE 1.14
Use this report to determine which files contain standards violations so they can be fixed.

Translating Layers (Windows Only)

Checking a drawing for standards violations is great for ensuring the files in the projects you create conform to your CAD standards, but working with drawings from a client or subcontractor can make this a bit more challenging, especially when it comes to layer standards. Over a set of drawings, you might use a few different dimension or text styles, but over those same drawings you could be working with dozens or hundreds of different layers.

AutoCAD provides a tool called the Layer Translator that allows you to map a single layer or a group of layers to a single layer based on your established CAD standards. Once a translation map is defined, it can be saved to a drawing or drawing standards file to reuse on other files from the client. This tool can also be useful in transitioning from an old to a new layer standard that your company is implementing. The Layer Translator can be displayed using the `laytrans` command.

Follow these steps to define a layer translation map and translate the layers in the current drawing to those defined by your CAD standards:

1. On the ribbon, click Manage tab ➢ CAD Standards panel ➢ Layer Translator (or at the command prompt, enter **laytrans** and press Enter). The Layer Translator dialog box (see Figure 1.15) is displayed.
2. In the Layer Translator, click Load in the Translate To area.
3. In the Select Drawing File dialog box, browse to and select the file that contains your CAD standards. Click Open. The layers in the file are populated in the Translate To list.
4. If your CAD standards file does not contain a layer you want to use as part of the layer translation map, click New. The New Layer dialog box (see Figure 1.16) is displayed.

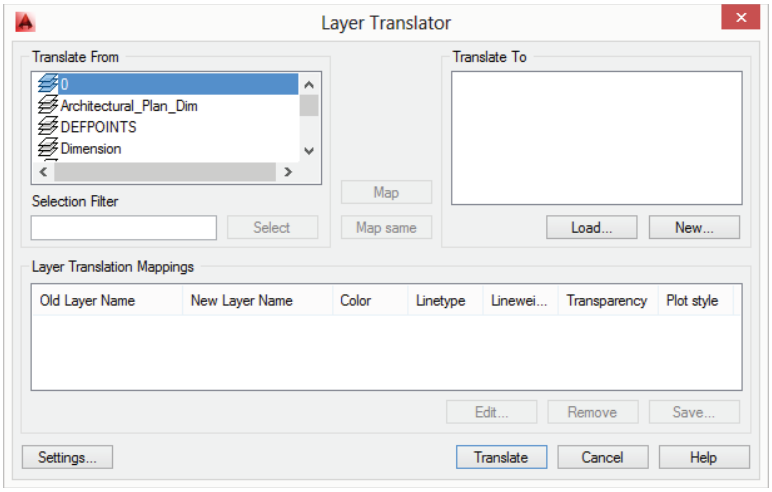


FIGURE 1.15 Translating layers between CAD standards

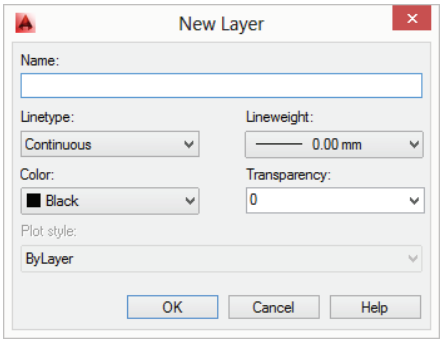


FIGURE 1.16 Defining a new layer to use as part of the layer translation map

5. In the New Layer dialog box, enter a name for the layer and define the new layer's properties. Click OK.

   The new layer is added to the Translate To list, but it is not actually created in the drawing until it is used as part of the layer translation mapping and the Translate button is clicked.

6. Click Map Same, adjacent to the Translate To area. This creates layer mappings for the layers that are named the same between the Translate From and Translate To areas.
7. In the Translate From area, select the layers that you want to map to a layer listed in the Translate To area. Press and hold Ctrl to select more than one layer in the Translate From area.

TIP If the layer list in the Translate From area contains a large number of layers, you can right-click the list and click Purge Layers to remove from the list the layers that are not being used in the current drawing. You can also enter a wildcard search in the Selection Filter text box to help you select layers listed in the Translate From area. For example, you can enter `*DIM*` to find all the layers that have a name containing the characters `DIM` some place in their names.

8. In the Translate To area, select the layer that the layers you selected in the Translate From list should be mapped to. Click Map.

   One or more layer mappings are added to the Layer Translation Mapping list, and the layers selected from the Translate From list are removed from the list since they cannot be mapped to multiple layers.
9. Click Settings. The Settings dialog box (see Figure 1.17) is displayed.

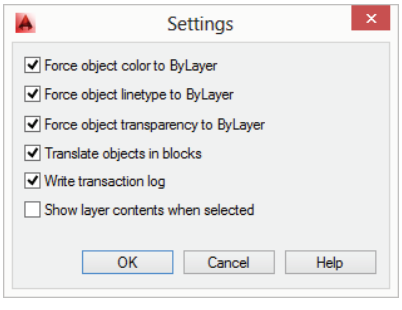
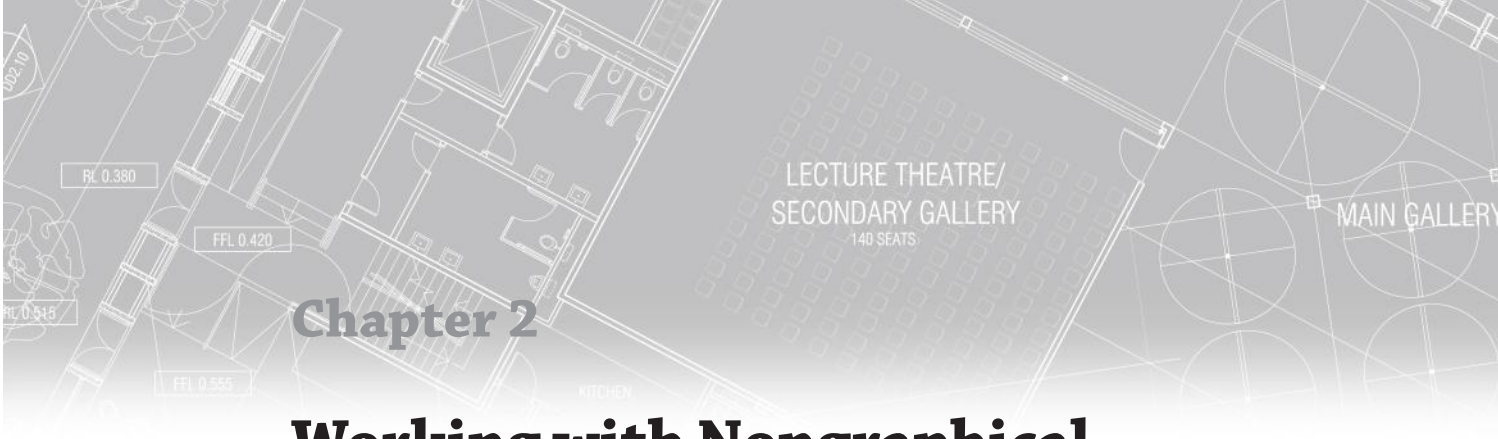


FIGURE 1.17 Changing the settings to use when translating layers

10. In the Settings dialog box, enable the options you want to use when translating the mapped layers. Click OK.
11. In the Layer Translator dialog box, click Translate.
12. In the Layer Translator - Changes Not Saved message box, click Translate And Save Mapping Information or Translate Only.
    - If you clicked Translate And Save Mapping Information, the Save Layer Mappings dialog box is displayed. Browse to a location and enter a name for the file. If needed, choose Standards (*.dws) or Drawing (*.dwg) from the Files Of Type drop-down list, and then click Save. After the translation mappings are saved, the layers are updated in the current drawing according to the layer mappings you created.
    - If you clicked Translate Only, the layers are updated in the current drawing based on the layer mappings you created and then the layer translation mappings are discarded.



Chapter 2

Working with Nongraphical Objects

Nongraphical objects, also known as *named objects*, are objects that are stored in a drawing but that are not visually part of your design in model space or paper space. They do affect the appearance of the linework and annotations visible in a drawing, control which objects are displayed, organize plot settings for outputting a layout, and much more. The nongraphical objects you might have to work with in a drawing include the following:

- Blocks
- Detail view styles
- Dimension styles
- Layers
- Layouts
- Linetypes
- Materials
- Multileader styles
- Multiline styles
- Plot styles
- Render presets
- Section view styles
- Table styles
- Text styles
- User coordinate systems (UCSs)
- Visual styles
- Viewports
- Views

Chances are, you've worked with many of these nongraphical objects and are already familiar with them. This chapter covers creating and managing the four most commonly used nongraphical objects: layers, text styles, dimension styles, and table styles. This chapter also explores the other nongraphical objects from the list and how you can work with them.

Standardizing the Names of Nongraphical Objects

The name of a nongraphical object is important to you and others who work in the drawings based on your company's CAD standards. Just as when you are naming projects and files, you should create meaningful names for your nongraphical objects so that you can quickly identify which object you need to work with and when. At a minimum, you will want to establish and use a naming standard for the layers in your drawing, since you can end up with dozens or even hundreds of different layers being used in a single drawing.

Establishing and following a naming standard helps ensure that the correct objects in a drawing are organized on the appropriate layers; as a result, they appear correctly in the output that you

eventually generate from a drawing. Utilizing standard names also makes it easy to identify which layers are used internally and allows your clients to efficiently use them with your drawings.

As with project- and file-naming standards, you can define your own naming standards or use those established by an industry body such as the American Institute of Architects (`www.aia.org`), National Institute of Building Sciences (`www.nibs.org`), Royal Institute of British Architects (`www.architecture.com`), or American National Standards Institute (`www.ansi.org`).

Layer Names

Layer names should be descriptive to give you control over the visibility and appearance of the objects in a drawing onscreen and during output. As part of a layer name, consider including the discipline, object classification/type, and the status of the objects on that layer. It is not uncommon to use layers to distinguish objects that are temporary construction lines from those of your design. Layers can also be used to distinguish walls that should be demolished from those that are to be constructed.

A basic layer name might contain the following information:

Discipline Often, a single letter represents the main discipline that will use the layer—for example, `A` for Architecture, `C` for Civil, `M` for Mechanical, or `S` for Structural.

Secondary Discipline Often, a single letter helps to provide an additional level of classification for the layer based on the designated main discipline—for example, `S` for site, `D` for demolition, or `I` for interior.

Major Classification Often, a grouping of three or four letters are used to identify the view or main purpose of the contents on the layer—for example, `-ANNO` for annotation, `-ELEV` for elevations, `-DETL` for details, and `-PLAN` for plans.

Minor Classification Level 1 Often, a grouping of three or four letters is used to further refine and specify the types of objects that might be on a layer—for example, `-DIMS` for dimensions, `-FURN` for furniture, `-NOTE` for notes, `-OBJT` for generic objects, and `-PATT` for hatches or fills. The minor classification might also contain a combination of alphabetic and numeric values to uniquely identify a layer within a drawing so that you can have more than one plan annotation layer—for example, `-A001` for annotation layer 001 or `-A020` for annotation layer 020.

Minor Classification Level 2 Often, a grouping of three or four letters can be used to further refine the use of the layer and which objects should be placed on it—for example, `-PRIM` for primary objects, `-OPNG` for opening, `-PIPE` for piping, and `-EQPM` for equipment.

Status Often, a single alphabetic or numeric value identifies the status of the objects on the layer or associates a phase with the objects—for example, `D` for demolish, `N` for new, or `1` for phase 1.

Figure 2.1 shows a possible structure for a layer name.

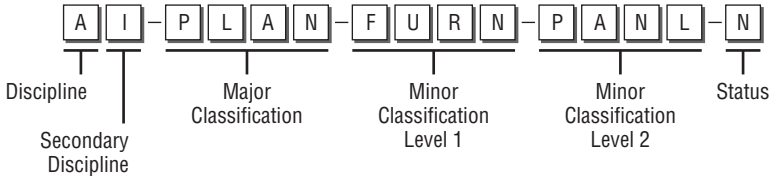


FIGURE 2.1: Possible layer-naming structure

Other Object Names

Unlike layers, the other nongraphical objects in a drawing often do not use a rigid naming structure. In most cases, these names are driven individually by each company because, compared to layers, they do not have as great of an impact on the ability to output a drawing. When you name the nongraphical objects that are not layers in a drawing, follow these suggestions:

- Keep names short, about 31 characters, so that they fit nicely into the list boxes and drop-down lists that are used by the dialog boxes and other areas of the Autodesk® AutoCAD® user interface.
- Be descriptive in your names and use abbreviations whenever possible. For example, the name DR32-90L is much more descriptive than the name D1. A new drafter will have a better chance of remembering that the block named DR32-90L represents a door that opens to the left and has an opening 32″ wide.
- Append the height of the text or dimension scale, such as 1_4 for 1/4″ height text or 96 for a scale of 1/8″ = 1′-0″, to each of your text or dimension style names if you are not using annotation scaling. This will make it easier to identify the correct styles without having to open one of the style managers to see the current value.
- Be consistent and use as many of the layer descriptors as possible with your block names, annotation styles, and named views. Consider using `-PLAN` for plan views, `-NOTE` for text styles that are used for general notes, or `-SECT` as part of a block name so that you know it should be used with section details. For example, if you want to create two new blocks that represent a single 2″ pulley that can be mounted on a surface, you might want to use the block names `SP2M-PLAN` and `SP2M-SIDE` for the plan and side views, respectively.

Renaming Nongraphical Objects

You can rename nongraphical objects using the dialog box that you used to originally create the object, or you can use the `rename` command. The `rename` command displays the Rename dialog box. If you are updating your existing CAD standards or moving to a new set of standards, you can use the `-rename` command in a script file or custom program to automate in a single operation the process of renaming several nongraphical objects in the drawings that have already been created. I discuss scripts in Chapter 8, "Automating Repetitive Tasks."

Use the following steps to rename a nongraphical object in an existing drawing:

1. Do one of the following:
   - At the command prompt, enter **`rename`** and press Enter (Windows and Mac OS).
   - Click Format menu ➢ Rename (Mac OS).
2. When the Rename dialog box (see Figure 2.2) opens, select the type of object you want to rename from the Named Objects list.
3. From the Items list, select the object you want to rename.
4. Do one of the following
   - In the Rename To text box, enter the new name for the object and click Rename To (Windows).
   - Select the object a second time to display the in-place text editor, and enter the new name for the object (Mac OS).

5. Repeat steps 2–5 for each object you want to rename.
6. Click OK when you have finished renaming objects.

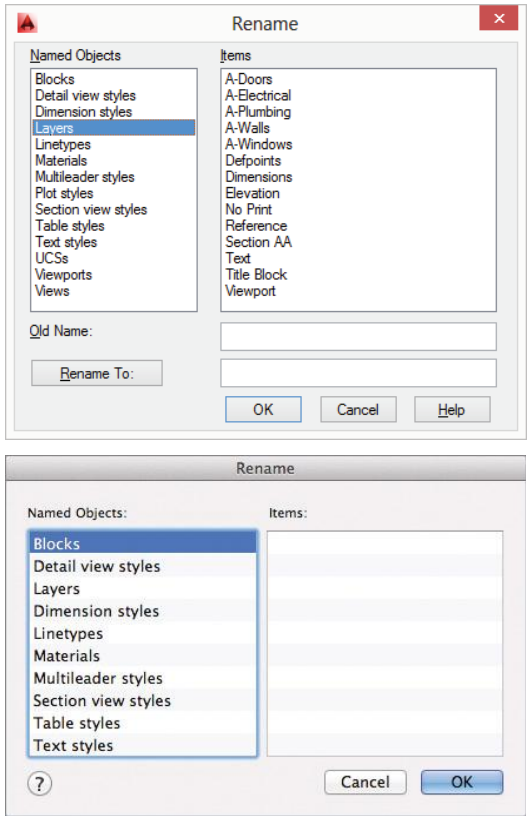


FIGURE 2.2: Renaming nongraphical objects

Managing Object Properties with Layers

Objects in a drawing have a number of properties in common with each other: color, linetype, lineweight, and several others. These are often referred to as an object's *general properties*. You can modify these properties individually using the Properties palette (Windows) or Properties Inspector (Mac OS) or let an object inherit values based on the layer in which it is placed.

I recommend allowing objects to inherit their properties from the layer in which they are placed; it is much easier to modify a single layer than it is to modify several thousands of objects. In addition, layers make it much simpler to control the visibility, locking, and plotting state of all related objects on a layer. An object's general properties must be set to `ByLayer` to allow it to inherit the property values of the layer it is on. Setting an object's property value directly overrides the layer's property value, which could have undesired effects when plotting or printing a drawing.

Layers are commonly created using the `layer` command. As a rule of thumb, don't allow individual users to create their own layers in each drawing, as doing so can introduce the following:

- Errors against the established CAD standards; the wrong layer name or incorrect property values
- Inefficiencies in your processes because of the amount of time it takes to define each layer that is needed

You can add all your commonly used layers to drawing template files so they are ready for use when the drawing is created. However, if a layer is not used it can inadvertently be purged from a drawing with the `purge` command. If a layer that was purged is needed later in the project, that layer will need to be re-created manually, inserted as a block that contains the missing layers, or re-created using an automated process. One of the simplest solutions to restoring standard layers that were purged from a drawing is to maintain a drawing file that contains the same layers as those in your drawing template files. Then, you can insert the drawing file with the `insert` command; all previously purged layers that were part of your drawing template file are restored and the layers that exist in both drawings are ignored.

In addition to adding your layers to a drawing template, consider using the `-layer` command in a script file or custom program to create the layers you need in a drawing. This approach has two benefits: you can use the script or program to create the layers in your drawing template file, and you can use it to reset the properties of layers if someone changes their values to be different from your CAD standards.

Setting the Default Properties for New Objects

When new objects in a drawing are created, they take on not only the current layer, but also a number of other values that are typically set to the value `ByLayer` or `0.0000`. You can adjust these properties in the General section of the Properties palette (Windows) or Properties Inspector (Mac OS) when no object is currently selected. You can also use the ribbon controls in AutoCAD on Windows. As an alternative, the system variables listed in Table 2.1 can be used to control the default property values assigned to new objects that are created with a command.

TABLE 2.1: System variables used to set default property values

SYSTEM VARIABLE	SETS
cecolor	Color
celtscale	Linetype scale
celtype	Linetype
celweight	Lineweight
clayer	Current layer
cmaterial	Material

TABLE 2.1: System variables used to set default property values *(CONTINUED)*

SYSTEM VARIABLE	SETS
cplotstyle	Plot style
elevation	Elevation in the Z direction
thickness	Object thickness

These system variables are helpful if you decide to create scripts or custom programs that create new objects or perform drawing setup tasks. I cover scripts in Chapter 8.

In most cases, you want to make sure that these variables are set to `ByLayer` or the equivalent value to make sure your new objects inherit the properties from the layer they are placed on. You can use the `setbylayer` command to reset an object's properties to `ByLayer`.

Creating and Managing Layers

You typically use the Layer Properties Manager (Windows) or Layers palette (Mac OS) to create new layers and edit existing ones in a drawing. When one of the interfaces is displayed, you click the Create New Layer (Windows) or New Layer (Mac OS) button and then set the properties in the Layers list for the new layer. Editing a layer is similar to the steps you take when creating a layer, except that you just need to click on the layer's row and in one of the properties on that row to begin editing the layer.

Follow these steps to create a new layer in AutoCAD on Windows:

1. On the ribbon, click Home tab ➢ Layers panel ➢ Layer Properties (or at the command prompt, enter **`layer`** and press Enter).
2. When the Layer Properties Manager (see Figure 2.3, top) opens, click Create New Layer.

TIP If a layer already exists with the property values close to the new layer you want to create, select the layer that you want to base the new layer on and then click Create New Layer.

3. Enter a name that follows your company's established CAD standards.
4. In the new layer's row, click one of the columns that represents the properties of the layer. The following explains what to do after clicking on the column:
   - On: Toggles the layer on or off. When set to Off, objects on the layer can still be selected using the `All` keyword at the `Select objects:` prompt, and the objects are regenerated when the drawing's display is updated.
   - Freeze: Toggles the freeze and thaw states of the layer. When set to Freeze, objects on the layer can't be selected using the `All` keyword at the `Select objects:` prompt, and the objects aren't regenerated when the drawing's display is updated.

- Lock: Toggles the lock state of the layer. The Lock setting restricts objects on the layer from being selected at the `Select objects:` prompt.
- Color: Displays the Select Color dialog box. Select a color value and click OK. For more information, see the section "Significance of Colors" later in this chapter.
- Linetype: Displays the Select Linetype dialog box. Select a loaded linetype and click OK. If the linetype you want to use is not loaded, click Load and load the layer first. For more information, see the section "Defining Appearance with Linetypes, Lineweights, and Transparency" later in this chapter.
- Lineweight: Displays the Lineweight dialog box. Select a lineweight and click OK. For more information, see "Defining Appearance with Linetypes, Lineweights, and Transparency."
- Transparency: Displays the Layer Transparency dialog box. Enter a new transparency value between 0 and 90, and click OK. For more information, see "Defining Appearance with Linetypes, Lineweights, and Transparency."
- Plot Style: Displays the Select Plot Style dialog box. From the Active Plot Style Table drop-down list, select the plot style file that you want to use for the current layout. Select one of the available plot styles to assign to the layer and click OK. (This option is available only if the drawing, drawing template, or drawing standards file is set up to use named plot styles.)
- Plot: Toggles the plottable state of the layer. Objects on the layer that is not plottable are displayed in the drawing window but are not part of the output when the drawing is plotted or printed.
- New VP Freeze: Toggles the freeze and thaw states of the layer when a new floating viewport is created on a named layout.
- VP Freeze: Toggles the freeze and thaw states of the layer in the current viewport.
- VP Color, VP Linetype, VP Lineweight, VP Transparency, and VP Plot Style: These properties can be changed just like the properties without the VP prefix previously described.
- Description: Double-click to edit the description of the layer. Adding a description can be helpful to those new to or unfamiliar with your CAD standards.

TIP Right-click the column headings in the Layer Properties Manager to control which columns you display. If there are properties that you commonly do not use, you can turn them off.

5. Repeat steps 2–4 for each layer you want to create.
6. Double-click one of the layers in the Layers list, select a layer from the Layers list, and click Set Current, or choose a layer from the Layer drop-down list on the Home tab ➢ Layers panel of the ribbon to set a layer as current.

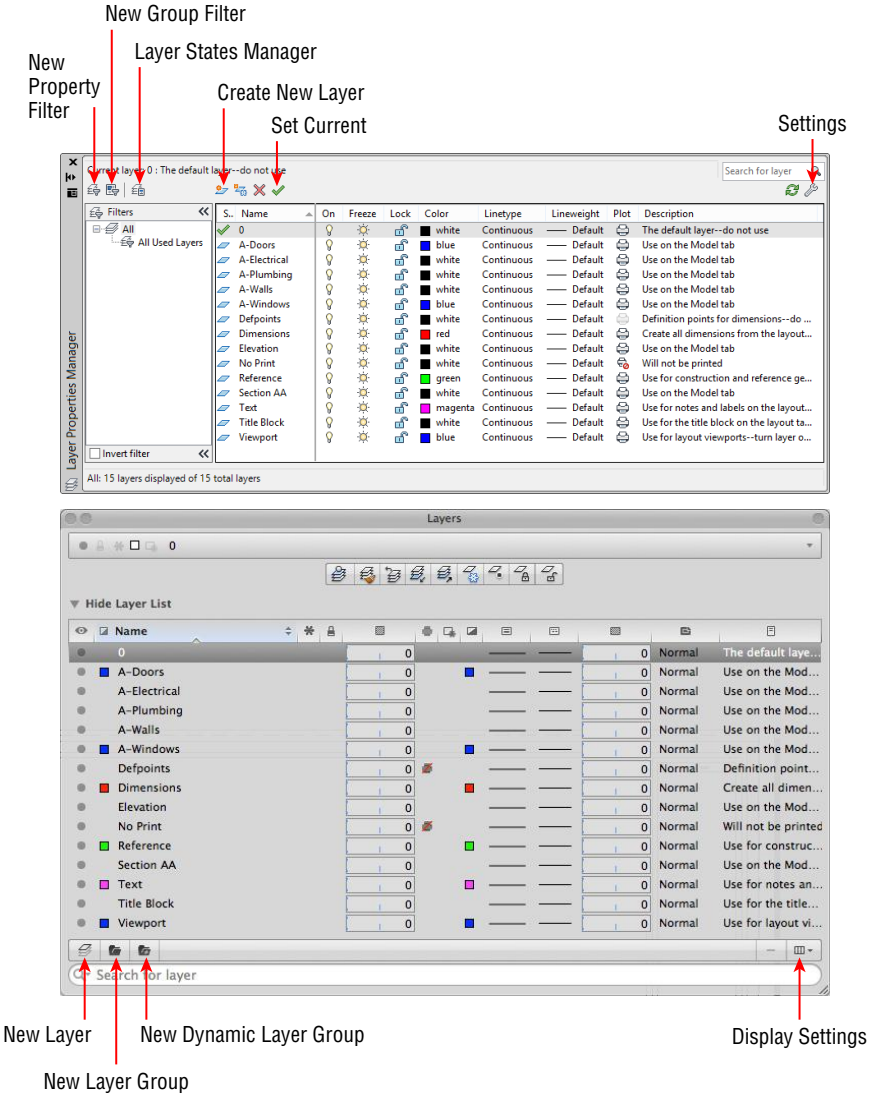


FIGURE 2.3: Organizing objects with layers

If you are using AutoCAD for Mac, follow these steps to create a new layer:

1. Click Format menu ➢ Layers (or at the command prompt, enter **`layer`** and press Enter).
2. On the Layers palette (see Figure 2.3, bottom), click New Layer.

TIP If a layer already exists with the property values that are close to those you need in the new layer you want to create, select the layer that you want to base the new layer on before clicking New Layer.

3. Enter a name that follows your company's established CAD standards.
4. With the new layer's row selected, do one of the following:
   - On the Layers palette, click one of the columns that represents the properties of the layer.
   - On the Properties Inspector palette, on the Layer Properties tab click one of the controls to edit the property's value.

The following explains what to do after you click on a column in the Layers palette or control in the Properties Inspector palette:

- On (Visibility): Toggles the layer on or off. When set to Off, objects on the layer can still be selected using the `All` keyword at the `Select objects:` prompt, and the objects are regenerated when the drawing's display is updated.

- Freeze: Toggles the freeze and thaw states of the layer. When set to Freeze, objects on the layer can't be selected using the `All` keyword at the `Select objects:` prompt, and the objects aren't regenerated when the display of the drawing is updated.

- Lock: Toggles the lock state of the layer. The Lock setting restricts objects on the layer from being selected at the `Select objects:` prompt.

- Color: Choose a color from the drop-down list or choose Select Color to display the Color Palette dialog box. Select a color value and click OK. For more information, see "Significance of Colors" later in this chapter.

- Linetype: Choose a linetype from the drop-down list or choose Manage to display the Select Linetype dialog box. Select a loaded linetype and click OK. If the linetype you want to use is not loaded, click Load and load the layer first. For more information, see "Defining Appearance with Linetypes, Lineweights, and Transparency" later in this chapter.

- Lineweight: Choose a lineweight from the drop-down-list. For more information, see "Defining Appearance with Linetypes, Lineweights, and Transparency."

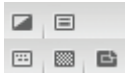
- Transparency: Click and drag the slide to specify a new transparency value. For more information, see "Defining Appearance with Linetypes, Lineweights, and Transparency."

- Plot Style: Choose a plot style from the drop-down list.

- Plot: Toggles the plottable state of the layer. Objects on the layer that is not plottable are displayed in the drawing window, but the objects are not part of the output when the drawing is plotted or printed.

- New VP Freeze/Freeze In New Viewports: Toggles the freeze and thaw states of the layer when a new floating viewport is created on a named layout.

- VP Freeze/Viewport Freeze: Toggles the freeze and thaw states of the layer in the current viewport.



- VP Color, VP Linetype, VP Lineweight, VP Transparency, and VP Plot Style: These properties can be changed just like the properties without the VP prefix previously described.

- Description: Edit the description of the layer. Adding a description can be helpful to those new to or unfamiliar with your CAD standards.

TIP Right-click the column headings in the Layers palette to control the display of which columns you want to display. If there are properties that you commonly do not use, you can turn them off.

5. Repeat steps 2–4 for each layer you want to create.
6. Double-click one of the layers in the Layers list or choose one from the Layers drop-down list at the top of the palette to set it current.

Significance of Colors

The use of color in a drawing serves two distinct purposes. The first is to make objects easy to distinguish from each other, and the second is to control the way the objects in your drawing are output. A majority of drawings use color-dependent plot styles. However, that does not mean they are plotted or printed in color, just that each unique color in a drawing can be used to control not only the color of an object when it is output, but also several other object properties. In addition to color, plot styles can override the linetype, lineweight, and transparency (or screening) values that are assigned to an object directly or that it inherits from the layer. Object and layer colors do not affect the output of objects when you are using named plot styles, the other type of plot styles that AutoCAD supports.

From the Layer Properties Manager (Windows) or Layers palette (Mac OS), clicking the Color column of a layer allows you to display the Select Color dialog box or Color Palette (see Figure 2.4), respectively. This interface allows you to select from one of the AutoCAD index colors (255 unique colors), a true color value, or a color from one of the installed color books. The AutoCAD index colors are the most commonly used for both layers and objects because these values directly map to values in a color-dependent plot style (CTB) file. If you need to set an object's color directly, select the object and use the Color property on the Properties palette (Windows) or Properties Inspector (Mac OS).

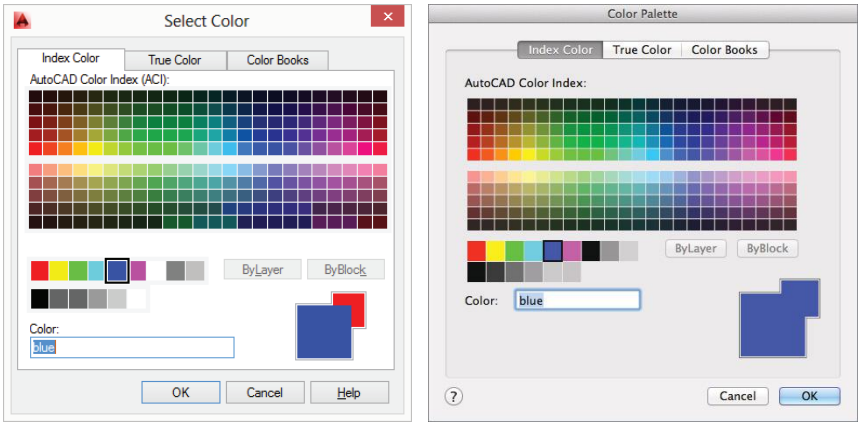


FIGURE 2.4: Setting the color of a layer

Defining Appearance with Linetypes, Lineweights, and Transparency

Color is just one property of an object that affects how it appears onscreen and when it is output; the linetype, lineweight, and transparency properties also impact the way objects appear. These properties can be set by object using the Linetype, Lineweight, and Transparency properties on the Properties palette (Windows) or Properties Inspector (Mac OS). However, as I mentioned earlier, it is much more effective to let objects inherit property values from the layer they are placed on.

Linetypes

Linetypes help distinguish objects that make up your design from those that are used to annotate it by using gaps, dashes, and text in your linework. Common uses for linetypes are to indicate center lines that pass through the center of a circle or an arc, to show features that are behind or beyond an object with hidden or dashed lines, or even to designate where a utility line runs through a property by displaying text within the linework.

The size of the dashes, gaps, and text in a linetype is determined by the linetype definition and two scale factors:

- The global scale factor, which is set by the `ltscale` system variable
- The linetype scale assigned directly to an object

The global scale factor is often equal to or half the drawing scale that will be used to output the drawing. If your viewports are set to a scale of 1/4" = 1'-0", your drawing scale would be a factor of 48, which is calculated by the math statement of (1/4) × 12. Once you know the drawing scale, divide it by 2, and that will give you the range in which your global linetype scale should be—in this example, that range would be 24 to 48.

The predefined linetypes that come with AutoCAD are stored in the `acad.lin` and `acadiso.lin` files. You can create your own linetype definitions and store them in the LIN files that come with AutoCAD, or you can create your own LIN files. Creating custom linetype definitions is covered in Chapter 9, "Defining Shapes, Linetypes, and Hatch Patterns." Linetypes must be loaded into a drawing with the Linetype Manager before they can be used. See Figure 2.5; the Windows version is displayed on top, and the Mac OS version is on the bottom. Once loaded, a copy of the linetype definition is stored in the drawing. When defining the layers in your drawing template files, only load the linetypes that are needed.

Lineweights

By default, all linework in a drawing is displayed as a single pixel in width onscreen unless the object is a polyline with a specified width or you use lineweights. Both layers and objects have a Lineweight property, which can be used to control the width or thickness of the linework for the objects in a drawing. Controlling the width of the linework allows you to emphasize the walls of a building or the edges of the main elements in a design while putting less emphasis on dimensions and other annotation objects. The correct balance of lineweight widths used in a drawing can improve how objects within the design are communicated to a client or contractor.

Lineweights are commonly used to control the width in which an object should appear when plotted or printed, but they can also be used to affect how objects are displayed onscreen. If you

want to display lineweights onscreen, you can use the Show/Hide Lineweight option on the application's status bar or change the value of the `lwdisplay` system variable.

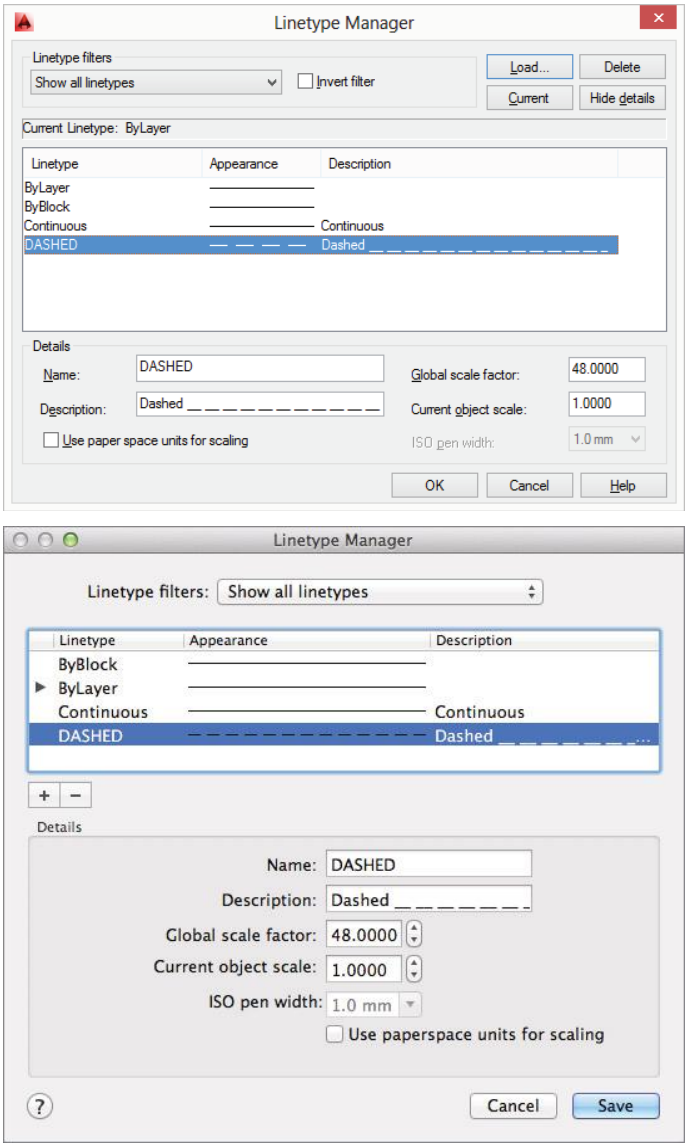


FIGURE 2.5: Managing linetypes and scales

The display of lineweights is different based on whether you are working on the model or in a named layout. The following explains the differences in the way lineweights are displayed.

Model Layout Independent of the current zoom scale, lineweights are measured in pixels. No matter the current zoom scale of the current view, the width of the linework never increases or decreases when zooming in or out.

Named Layout Lineweights are displayed based on the current zoom scale. Just like placing a magnifying glass over a sheet of paper, the magnification of the drawing makes the linework appear thicker; the same happens when zooming in or out on a layout.

You can modify the settings that affect the display of lineweights in a drawing by using the Lineweight Settings dialog box. See Figure 2.6; the Windows version is displayed on the left, and the Mac OS version is on the right. The Default (Windows) or New Layer Default (Mac OS) drop-down list controls the lineweight used by the Default value of the Lineweight property for layers and objects, which can also be set with the `lwdefault` system variable. Use the Adjust Display Scale (Windows) or Preview Scaling (Mac OS) slider to control the lineweight scale of all lineweights on the Model layout. The other area that may have some significance is the Units For Listing (Windows) or Units (Mac OS) section, which controls the values in which lineweights are expressed in the AutoCAD user interface: millimeters (mm) or inches (in). Units for listing lineweights can also be controlled with the `lwunits` system variable.

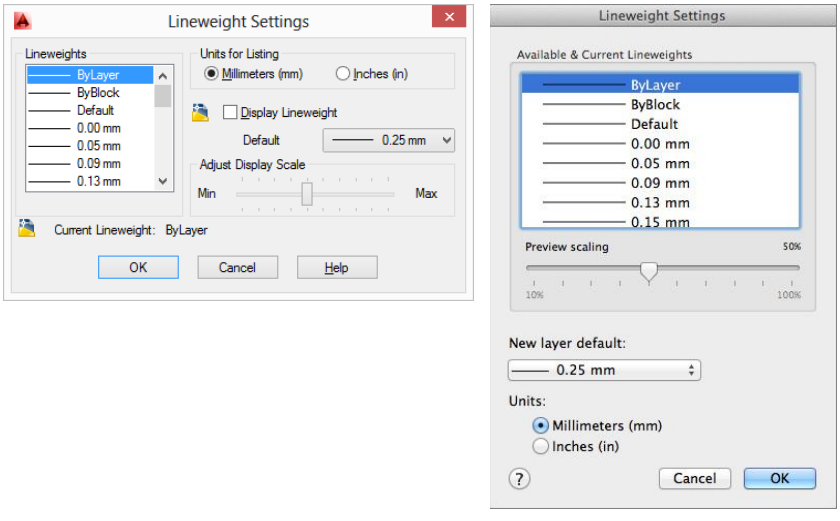


FIGURE 2.6: Controlling the display of lineweights

TRANSPARENCY

Transparency gives you the ability to see through to what is below or beyond an object. Objects can have a transparency value of 0 (fully opaque/solid) to 90 (nearly fully transparent). Unlike lineweights, the display of transparency is much more useful onscreen in a complex drawing, but it can have an impact on the display of your drawing based on your workstation's hardware. You can use the Show/Hide Transparency option on the application's status bar or change the value of the `transparencydisplay` system variable.

Controlling Output with Plot Styles

While layers and object properties control much of the way objects are displayed onscreen, you can use plot styles to override those values to alter the way objects are plotted or printed. Plot styles can also be used to affect the way objects appear onscreen if the Display Plot Styles option is enabled in the Page Setup dialog box.

Plot styles are available in two styles: color-dependent and named. When you are using color-dependent plot styles, the name of the plot style is fixed based on the color value assigned to an object or the layer in which an object is placed. Assigning a named plot style to a layer or object is similar to assigning a linetype: first you specify which named plot style file you want to use, and then you specify the plot style you want to assign to the layer or object. For more information on plot styles and how to create them, see Chapter 1, "Establishing the Foundation for Drawing Standards."

Accessing Layer Settings

Besides the properties for a layer that are accessible from the Layer Properties Manager (Windows) or Layers palette (Mac OS), several other settings affect the way layers look or behave in the drawing window. In AutoCAD for Windows, you can use the Layer Settings dialog box (see Figure 2.7) to change the current layer settings, but you can also use a few system variables to change some of these settings. In the dialog box, the settings with a small drawing file icon next to them (the blue-and-yellow sheet of paper) indicate that the setting is stored with the drawing.

In AutoCAD for Mac, there is no equivalent to the Layer settings dialog box (many of the same features in the Windows product are not available on Mac OS), but there are a few settings that you can access using system variables and commands to change your experience.

The Layer Settings dialog box in AutoCAD on Windows can be displayed by doing the following:

1. On the ribbon, click Home tab ➢ Layers panel ➢ Layer Properties.
2. In the Layer Properties Manager, click Settings, located near the upper-right corner.

Fading Objects on Locked Layers

Locking a layer ensures that the objects on that layer are not accidentally selected when editing other objects in a drawing, but they still can be used as reference geometry. To help make it easy to identify when a layer is locked, AutoCAD fades the objects on a locked layer so they are less prominent in the foreground. You can toggle this feature on and off, as well as control the amount objects are faded. By default, objects on locked layers are faded by 50 percent.

To change the amount a layer is faded, you can do one of the following:

- In the Layer Settings dialog box (see Figure 2.7), in the Isolate Layer Settings section, choose Lock And Fade, and then drag the Locked Layer Fading slider. If the slider is not enabled, click the Locked Layer Fading toggle (the stack of three papers with a lock icon) and then drag the slider. Click OK (Windows).
- On the ribbon, select Home tab ➢ Layers panel, click the panel's title to expand the panel, and drag the Locked Layer Fading slider. If the slider is not enabled, click the Locked

Layer Fading toggle (the stack of three papers with a lock icon) and then drag the slider (Windows).

- At the command prompt, enter **`laylockfadectl`** and press Enter. Enter a new fade value and press Enter. Entering a negative value disables the fading of objects on a locked layer (Windows and Mac OS).

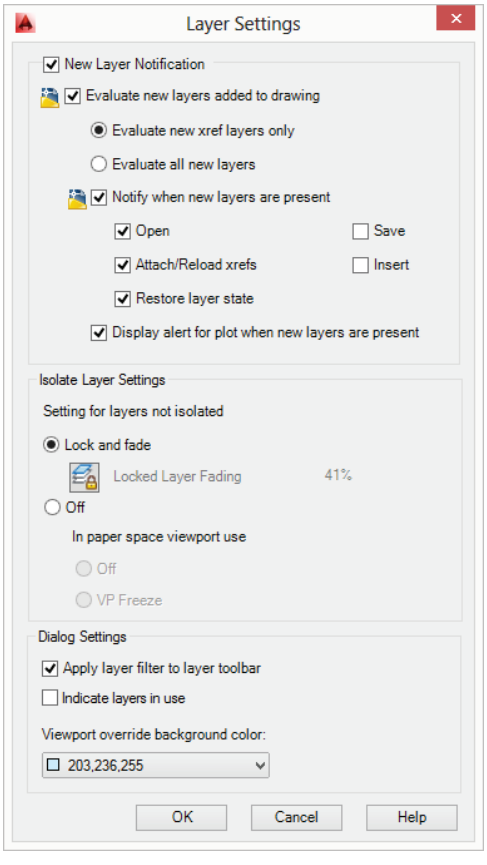


Figure 2.7: Changing layer settings affects the way you work with layers in the drawing window and user interface.

Isolating Objects; Turning Off or Locking

Isolating layers allows you to quickly turn off or lock layers, work with objects on other layers in the drawing, and then quickly restore the previous state of the layers in the drawing. Layers can be isolated with the `layiso` command, and isolation can be reversed (unisolated) with the `layuniso` command. You can control the default isolation mode that the `layiso` command uses.

To change the isolation mode of the `layiso` command, you can do one of the following:

- In the Layer Settings dialog box (see Figure 2.7), in the Isolate Layer Settings section, choose Lock And Fade to lock the layers when using `layiso` or click Off to turn them off

instead. When you click Off, you have the option to turn the layer off or to freeze it in the current viewport when using `layiso` in a floating viewport. Click OK (Windows).

- At the command prompt, enter **`layiso`** and press Enter. Use the Settings option and specify the mode to use (Windows and Mac OS).

Evaluating and Reconciling New Layers

It is not uncommon to need a new layer here and there that deviates from your company's CAD standards, but new layers can affect the way a drawing is plotted or printed based on how that layer was defined. AutoCAD on Windows and Mac OS has a feature known as *New Layer Notification*, but it works slightly differently on the two platforms. On Windows, when the New Layer Notification feature is enabled and a new layer has been added to a drawing, a notification balloon (see Figure 2.8) is displayed, and a new layer filter named Unreconciled New Layers is created in the Layer Properties Manager. By default, the notification of new layers happens when an external reference is being attached, when a drawing is opened, when an xref is attached/reloaded, and when a layer state is restored.

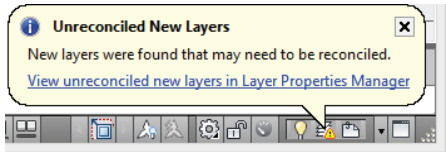


Figure 2.8: Unreconciled layer notification balloon

On Mac OS, the feature is disabled out-of-the-box, so it must first be enabled. The notification balloon is not available on Mac OS, but the Unreconciled New Layers filter is created in the Layers palette when the feature is enabled and the correct conditions are met.

You can modify the settings of the New Layer Notification feature from the Layer Settings dialog box (see Figure 2.7) in AutoCAD on Windows, or with the following system variables on both Windows and Mac OS:

`layereval` Controls whether the layers in a drawing should be evaluated only when an xref is attached, or when a new layer is created in addition to when an xref is attached.

`layerevalctl` Disables or enables the New Layer Notification feature.

`layernotify` Specifies which drawing editor events display an alert message when `layerevalctl` is set to 1.

To reconcile a layer in AutoCAD on Windows, you need to perform the following steps:

1. On the ribbon, click Home tab ➢ Layers panel ➢ Layer Properties.
2. On the Layer Properties Manager, select the Unreconciled New Layers filter from the Filter Tree located on the left side. (If the Filter Tree is not displayed, right-click in the Layers list and click Show Filter Tree.)
3. In the Layers list, select the layers you want to reconcile and then right-click. Click Reconcile Layer.

4. Save the drawing. If saving a drawing is one of the events that triggers the Unreconciled New Layers notification, the balloon will be displayed each time a drawing is saved if you do not reconcile (or remove) all of the offending layers from the drawing.

Reconciling a layer in AutoCAD for Mac OS requires actions similar to those used on Windows. You can reconcile a layer on Mac OS using the following procedure:

1. Click Format menu ➢ Layers.
2. In the Layers palette, expand the Unreconciled New Layers filter from the Layers list. If the filter is not displayed, click Display Settings in the lower-right corner of the Layers palette and click Show Layer Groups; the option should now be checked. To make sure Unreconciled Layers is also checked, click Display Settings ➢ Show Automatic Groups ➢ Unreconciled Layers.
3. In the Layers list, select the layers you want to reconcile and right-click. Click Reconcile Layer.
4. Save the drawing.

For more information on the New Layer Notifications feature and reconciling layers, see the AutoCAD Help system.

Controlling the Layer Properties Manager and Ribbon Controls (Windows Only)

There are several settings that you can use to alter your experience with the Layer Properties Manager and the Layer list on the ribbon. These settings can be found under the Dialog Settings section of the Layer Settings dialog box (see Figure 2.7). The settings that can be changed:

Apply Layer Filter To Layer Toolbar (or Layer Drop-Down List on the Ribbon) This option aligns the list of layers displayed in the Layer Properties Manager with the Layer control on the Layer toolbar or Layers panel of the ribbon. This is helpful when working with drawings that contain hundreds of layers.

Indicate Layers In Use Use this option to control whether the icon in the Status column is updated to reflect if at least one object is placed on that layer in the drawing. This option can also be controlled with the `showlayerusage` system variable.

Viewport Override Background Color This option enables the highlighting of layers with viewport overrides in the Layers list of the Layer Properties Manager, or those in the Layer drop-down list on the Layer toolbar or Layers panel of the ribbon.

In addition to the settings under the Dialog Settings section of the Layer Settings dialog box, you can right-click the Layers list of the Layer Properties Manager to control the display of the Filter Tree and whether filters appear in the Layers list. You can also right-click a column heading in the Layers list to control which columns are displayed, or click and drag a column heading to reorder the columns. Clicking and dragging between columns adjusts the width of the column to the left.

Controlling the Layers Palette (Mac OS Only)

The Layers palette in AutoCAD for Mac doesn't have as many settings that control its behavior as the Layer Properties Manager in AutoCAD on Windows, but there are a few settings that can improve your experience. These settings are available from the Settings menu, which you can display by clicking the Display Settings button located in the lower-right corner of the Layers palette. These settings are available on the Settings menu:

Show Layer Groups This option controls the display of layer groups within the Layers list. You can also control where the layer groups are displayed: at the top or bottom of the Layers list.

Show Empty Groups Use Show Empty Groups to display or hide layer groups that do not have any layers in them.

Show Xref Layers This option controls the display of layers from attached xrefs in the Layers list.

Show Automatic Groups Show Automatic Groups controls the display of auto-generated layer groups based on specific criteria: all used layers, attached external references, layers with viewport overrides, and unreconciled layers.

View Options This option controls which columns are displayed in the Layers list.

You can also right-click a column heading to control which columns are displayed, or click and drag a column heading to reorder the columns. Clicking and dragging between columns adjusts the width of the column to the left.

Grouping and Filtering Layers

Filters in the Layer Properties Manager (Windows) or Layers palette (Mac OS) allow you work with groupings of layers much like you would an individual layer. AutoCAD creates a few layer filters dynamically based on events that happen in the drawing window, such as attaching an xref or creating a viewport override on a layer. You can also create one of two types of layer filters based on your own conditions:

Group Filter Using a group filter, you can create a static grouping of layers that are manually selected.

Dynamic or Property Filter Using a dynamic or property filter, you can create a grouping of layers defined by the properties they have in common.

You can create a group filter and add layers to it by taking the following steps:

1. Do one of the following:
   - On the Layer Properties Manager, click New Group Filter (Windows).
   - On the Layers palette, click New Layer Group (Mac OS).
2. Enter a descriptive name for the new group and press Enter.
3. Drag and drop layers from the Layers list onto the new group to create an association between the layers and group.

A property filter in AutoCAD on Windows can be created by following these steps:

1. On the Layer Properties Manager, click New Property Filter.
2. When the Layer Filter Properties dialog box (Figure 2.9, top) opens, enter a descriptive name in the Filter Name text box.
3. In the Filter Definition grid, set the properties you want to filter on. Click OK.

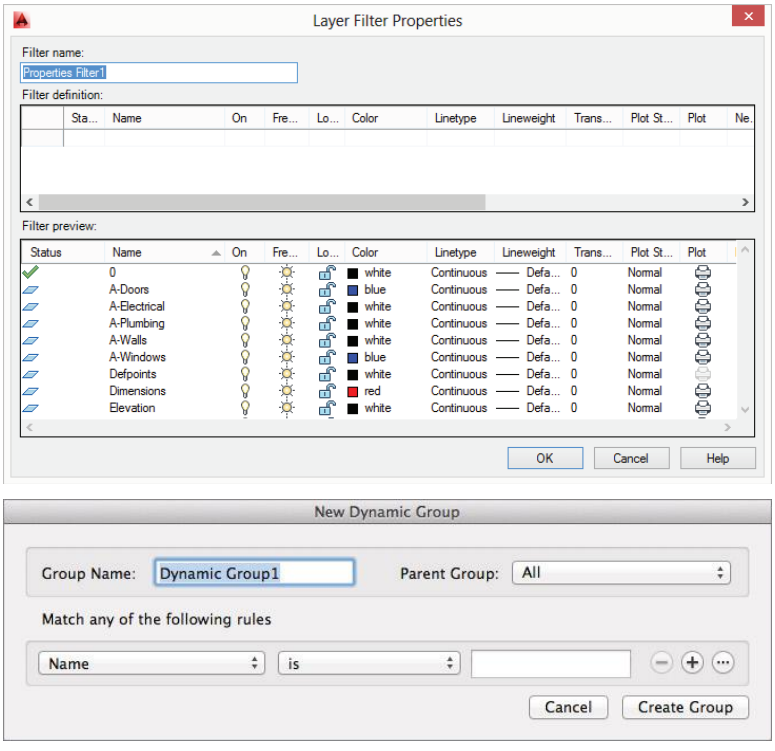


FIGURE 2.9: Creating dynamic filters based on layer names and properties

If you are using AutoCAD for Mac, do the following to create a dynamic filter:

1. On the Layers palette, click New Dynamic Layer Group.
2. When the New Dynamic Group dialog box (Figure 2.9, bottom) opens, enter a descriptive name in the Group Name text box.
3. In the Match Any Of The Following Rules section, set the properties you want to filter on.
4. Optionally, click the + (plus) button to add properties to the filter.
5. Click Create Group.

Creating and Using Layer States

Layer states allow you to take a snapshot of the current layers in a drawing. The snapshot includes the layers that currently exist in the drawing and their current property values. Any new layers added to a drawing are not automatically added to an existing layer state because they did not exist when the layer state was created. When a layer state is restored and new layers exist in the drawing, you have the option to turn off any layers that weren't saved with the layer state.

After you make changes to the layers in the drawing for editing, plotting, or display purposes, you can return the layers to their previous properties by restoring a saved layer state. Layer states can be beneficial when you receive a drawing from a client and want to make sure that the layers are just like they were when you received the drawing before sending it back to the client. You can create layer states in AutoCAD on Windows by using the Layer States Manager dialog box (see Figure 2.10) or with the `State` option of the `-layer` command. The Layer State Manager does not exist in AutoCAD for Mac, so you need to use the `State` option of the `-layer` command.

The following steps explain how to create a layer state in AutoCAD on Windows:

1. On the ribbon, click Home tab ➢ Layers panel ➢ Layer States drop-down list ➢ Manage Layer States (or at the command prompt, enter **`layerstate`** and press Enter).
2. When the Layer States Manager (Figure 2.10) opens, click New.
3. In the New Layer State To Save dialog box, enter a name in the New Layer State Name text box. Optionally, enter a description. Click OK.
4. Optionally, click Edit to edit the layers and the properties that are being saved with the layer state. Make the edits in the Edit Layer State dialog box and click OK.
5. Click Save and then click OK.

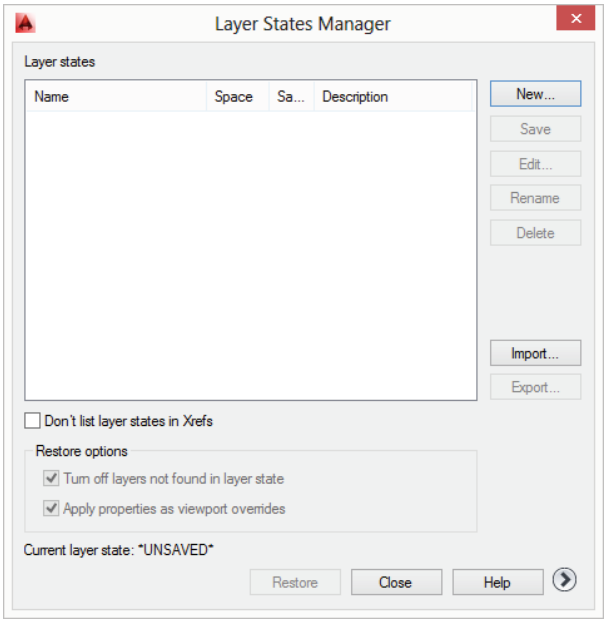


FIGURE 2.10: Saving layer states allows you to later restore the visibility and properties of the layers in a drawing.

If you are using AutoCAD for Mac or want to create a layer state from the command prompt in AutoCAD on Windows, do the following:

1. At the command prompt, enter **-layer** and press Enter.
2. At the `Enter an option [?/Make/Set/New/Rename/ON/OFF/Color/Ltype/LWeight /TRansparency/MATerial/Plot/PStyle/Freeze/Thaw/LOck/Unlock/stAte/ Description/rEconcile]:` prompt, type **state** and then press Enter.
3. At the `Enter an option [?/Save/Restore/Edit/Name/Delete/Import/EXport]:` prompt, type **save** and then press Enter.
4. At the `Enter new layer state name:` prompt, type a name for the new layer state and then press Enter.
5. At the `Enter states to change [On/Frozen/Lock/Plot/Newvpfreeze/Color/line-Type/lineWeight/TRansparency/plotStyle]:` prompt, type a property to change and then press Enter, or simply press Enter if you don't want to make any changes to the new layer state.
6. Press Enter again to end the `-layer` command.

A layer state after it is saved can be restored by doing one of the following:

- In the Layer States Manager, select the layer state you want to restore and click Restore (Windows).
- On the ribbon, click Home tab ➢ Layers panel ➢ Layer States drop-down list and select the layer state you want to restore (Windows).
- At the command prompt, type **-layer** and press Enter. At the `Enter an option` prompt, type **state** and press Enter. At the `Enter an option` prompt, type **restore** and press Enter. At the `Enter name of layer state to restore or [?]:` prompt, type the name of the layer state to restore and press Enter. Press Enter again to end the `-layer` command (Windows and Mac OS).

Creating and Managing Annotation Styles

Annotation styles play a significant role in the communication of your design to those who will sign off on the project or be involved in manufacturing. AutoCAD supports four primary annotation styles that affect the appearance of text, dimension, table, and multileader objects. The following sections explain the basics of creating and editing these annotation styles.

Text Heights

The height at which any text should be created is based on where the text will reside: model space or paper space. Text created in model space (using the Model tab) is commonly scaled up because the objects represent real-world objects that are being designed and drawn at full scale and then are scaled down when plotted or printed to fit on a sheet of paper.

Text at a height of 3/16″ (or 0.1875 inches) is very small in model space if you normally use a plot scale of 1/8″ = 1′-0″ or 100:1 for your drawings. The text would be plotted at about

0.0019 inches high, or basically a dot on the drawing. To get an acceptable size for your text in model space, you take the text height you want the final text to appear, say 3/16", and multiply it by the expected plot scale (1/8" = 1'-0" is equal to 96). So you take (3/16) × 96 to calculate the final text height of 18" to be used for text in model space. On the other hand, 18" is way too large for text in paper space. Here, you would use the actual text height of 3/16" since you commonly plot or print a layout at a scale of 1:1.

Things can get complicated even further if you plot parts of your drawing at different scales since your text would not look the same when plotted at a scale of 1/4" = 1'-0" as it would at 100:1. The issue of dealing with multiple plot scales in a drawing can be addressed with one of two solutions:

- Use different layers and create multiple text objects at different heights; then control which objects should be displayed in a viewport at a specific scale or when the drawing is plotted. Managing multiple layers and annotation objects was once the only choice, but it is still the common choice for some companies.
- Use annotative scaling to dynamically scale text up or down based on the scale at which the object is being viewed through a viewport or plotted. I discuss annotative styles and annotation scaling in the section "Annotative Styles and Annotation Scaling" later in this chapter.

Text Styles

Text styles are used to define how text within a drawing will look or behave when created, but not what the contents of the text will be. For example, a text style defines which font will be used to control the appearance of the characters within a text object, if the text is bold or italicized, the default height of the text, or if the object will be annotative when a new text object is created.

Text styles in AutoCAD can be defined to use one of two font types:

Shape (SHX) Fonts Shape fonts are optimized for and only work with AutoCAD. Shape fonts are defined through a series of vectors, which make them more efficient than TrueType fonts. Shape fonts are also used when specifying a Big Font file to use for Asian languages. You can create your own custom fonts or characters by defining them in a Shape (SHP) file. I discuss creating shapes in Chapter 9.

TrueType (TTF) Fonts TrueType fonts are common to the operating system and other applications on a workstation. They offer a wider range of looks than the SHX files do, but they can impact the performance of zooming and object selection in AutoCAD.

Make sure the fonts that you choose to use are easy to read; at the end of the day, annotation is about communicating your design, not for winning awards at an art show. The fonts you pick can affect sharing or exchanging of drawings, too. If a font is not available on a workstation that opens the drawing, a substitute font is used and the text might not appear properly. When a font is not available, AutoCAD replaces the missing font with the one specified by the `fontalt` system variable.

Creating and Modifying Text Styles

Text styles are created and modified using the Text Style dialog box, which is displayed with the `style` command. The process for creating and modifying a text style is similar on Windows

and Mac OS. There are some minor differences because of the way the dialog boxes are laid out. After a text style has been created with the Text Style dialog box, you use the dialog box again to make changes by selecting the style you want to edit from the Text Styles list and changing the properties of that style. Once the property changes are made, you click Apply to save the changes. If you want to create or modify text styles using scripts or custom programs, you can use the `-style` command.

The following explains how to create a text style in AutoCAD on Windows:

1. On the ribbon's Annotate tab ➢ Text panel, click the panel-launcher button located to the right of the Text panel's title (or at the command prompt, enter **`style`** and press Enter).
2. When the Text Style dialog box (Figure 2.11, top) opens, click New. Enter a name for the new text style and click OK.
3. Select a TTF or SHX file from the Font Name drop-down list. Choose an option from the Font Styles drop-down list as needed. If you need to create a style with a Big Font, select an SHX font and click Use Big Font. Then, specify a font from the Big Font drop-down list.
4. Enter a text height in the Height box. The value entered becomes the default text height when you create text objects. It also affects other annotation styles that use the text style. You can create annotative text, which I cover in the section "Annotative Styles and Annotation Scaling" later in this chapter.
5. In the Effects section, specify any of the options needed for your text style.
6. Click Apply.
7. Double-click the text style you want to make current. When no objects are selected, you can also set a text style as current from the Text Style drop-down list, which is available on the ribbon's Annotate tab ➢ Text panel.
8. Click Close.

If you are using AutoCAD for Mac, you can do the following to create a text style:

1. Click Format menu ➢ Text Style (or at the command prompt, enter **`style`** and press Enter).
2. When the Text Style dialog box (Figure 2.11, bottom) opens, click the + (plus) button located in the lower-left corner. Enter a name for the new text style and press Enter.
3. Select a TTF or SHX file from the Family list box. Choose an option from the Typeface list box as needed. If you need to create a style with a Big Font, select an SHX font and then select a font from the Asian Set list box.
4. Enter a text height in the Text Height box. The value entered becomes the default text height when you create text objects; it also affects other annotation styles that use the text style too. You can create annotative text as well, which I cover in the section "Annotative Styles and Annotation Scaling" later in this chapter.
5. Specify any of the effects in the Text Style Preview area as needed.
6. Click Apply.

7. Double-click the text style you want to make current. You can also set a text style as current from the Text Style drop-down list in the Annotation section of the Properties Inspector when no objects are selected.

8. Click Close.

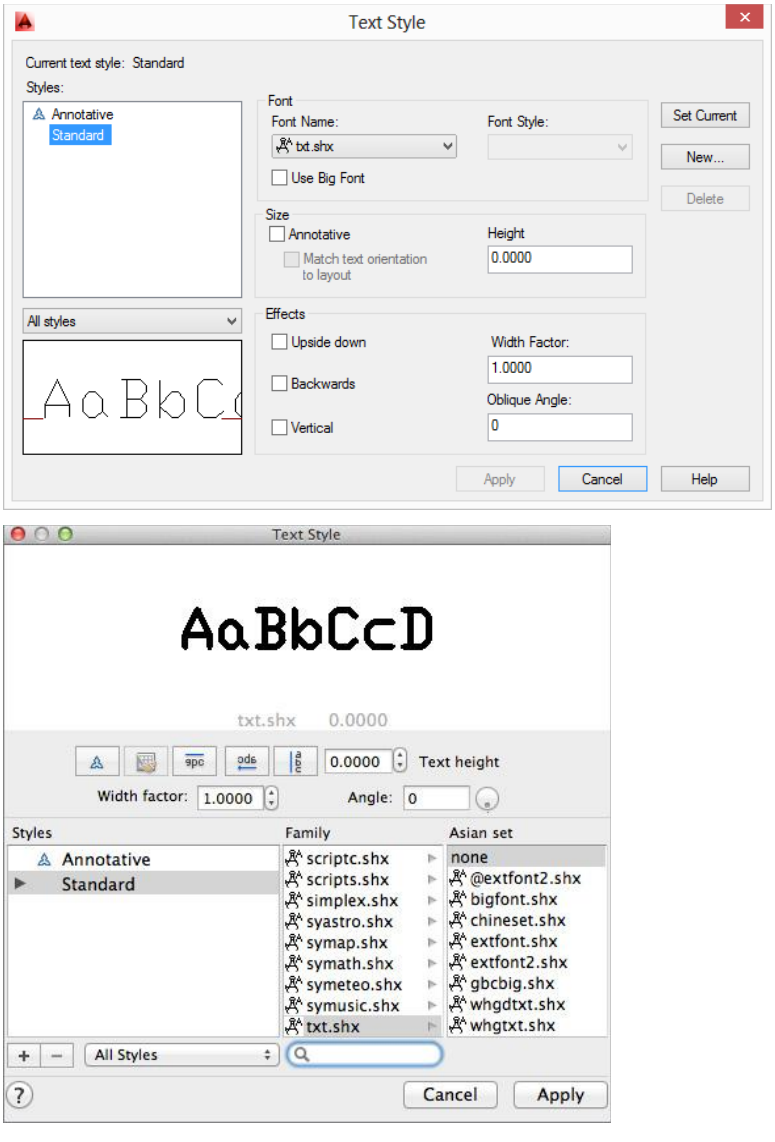


FIGURE 2.11: The appearance of text can be controlled with text styles.

Using a Text Style with Dimension, Table, and Multileader Styles

When you create a text style, you have the option to specify a default text height for use when you create new text objects. Setting a text height to a specific value also forces any dimension, table, or multileader styles that reference that text style to take on the text height to which the text style is set. Don't use a specific text height when you define text styles unless you wish to restrict flexibility in changing heights when creating text objects or using a text style with other annotation styles.

Dimension Styles

Dimension objects are made up of a number of different components, from arrowheads to extension and dimension lines to annotation objects with formatted values. All of the components that make up a dimension object are defined using dimension styles. Dimension styles allow you to control the text size and placement, dimension and extension line colors and linetypes, as well as a dimension scale, which is used to scale up or down the various components of a dimension so that they display correctly based on a drawing's plotted scale.

Dimension styles are created and modified using the Dimension Style Manager, which is displayed with the `ddim` command. If you want to create or modify dimension styles using scripts or custom programs, you will want to take a look at the `-dimstyle` command and the dozens of system variables that all begin with the letters `dim`.

TIP You can use the Compare button or option from the Dimension Style Manager to see the differences between two different dimension styles.

You can create a dimension style on Windows or Mac OS with these steps:

1. Do one of the following to display the Dimension Style Manager:
   - On the ribbon's Annotate tab ➢ Dimensions panel, click the panel-launcher button located to the right of the Dimensions panel's title (Windows).
   - Click Format menu ➢ Dimension Style (Mac OS).
   - At the command prompt, enter **`ddim`** and press Enter (Windows and Mac OS).

   Figure 2.12 shows the Dimension Style Manager as it appears in Windows (top) and Mac OS (bottom).
2. In the Dimension Style Manager, click New (Windows) or the + (plus) button (Mac OS).
3. In the Create New Dimension Style dialog box, enter a name for the new style.
4. Optionally, do the following:
   - Select an existing style to start with and check Annotative if you want to create an annotative style. I discuss annotative styles in the section "Annotative Styles and Annotation Scaling" later in this chapter.
   - Select an option from the Use For drop-down list to have the new dimension style apply only to a subset of dimension objects. Typically, keep the Use For drop-down list set to All Dimensions.

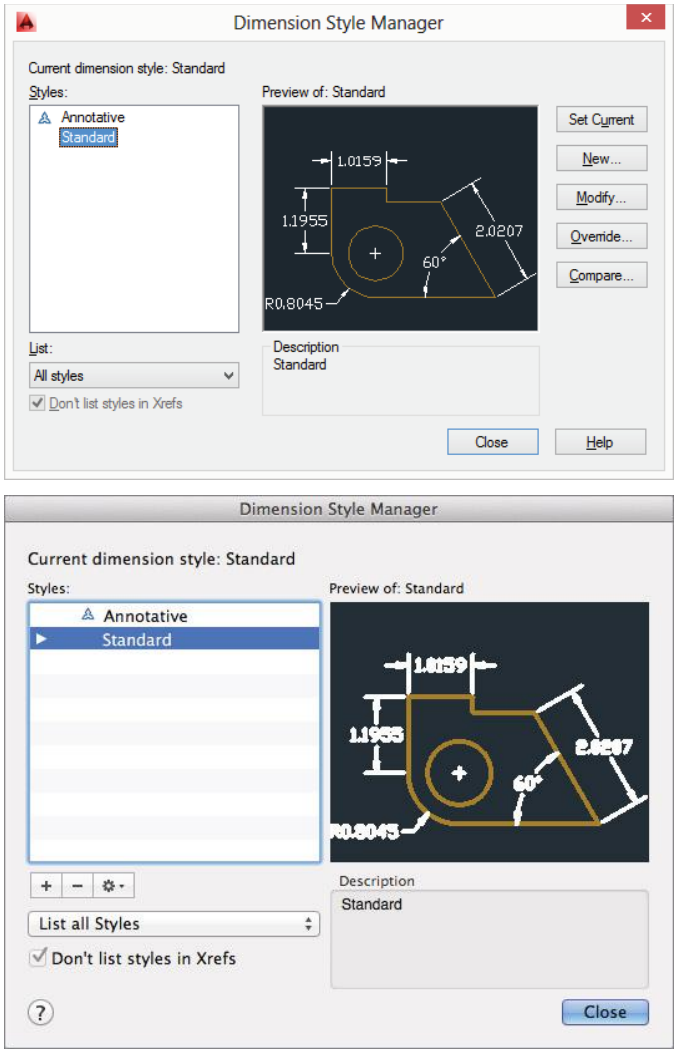


FIGURE 2.12: Creating and defining dimension styles

5. Click Continue.
6. When the New Dimension Style dialog box opens, start on the Lines tab and change the settings as needed.
7. Continue making changes on each tab in the dialog box.
8. Click OK to return to the Dimension Style Manager.
9. Right-click the dimension style you want to make current and click Set Current. When no objects are selected in AutoCAD on Windows, you can also set a dimension style as

current from the Dimension Style drop-down list on the Annotate tab ➢ Dimensions panel on the ribbon. When no objects are selected in AutoCAD for Mac, you can set a dimension style as current from the Dimensions Style drop-down list in the Annotation section of the Properties Inspector.

10. Click Close.

You can edit a dimension style by doing one of the following when the Dimension Style Manager is displayed:

- Select a dimension style from the Styles list box and click Modify (Windows).
- Select a dimension style from the Styles list box, click the Action button (the gear icon), and then click Modify (Mac OS).

Dimension overrides, which allow you to apply a temporary change to a style, can also be created. The changes based on the overrides are added to new objects that are created when the dimension style is current; existing dimensions are not affected by the override. You create an override to a dimension style much like you would modify a dimension style, except that instead of clicking the Modify button/option you click Override. You can also override the properties of individual dimension objects by using the Properties palette (Windows) or Properties Inspector (Mac OS).

Table Styles

Table styles are used to control the direction in which content flows for a table, as well as the appearance of the grid border lines and textual content that appears in table cells that are assigned a given style. A table style can also contain cell styles that are used to group general formatting, text, and border settings for use in a table.

AutoCAD on both Windows and Mac supports the creation of tables and the use of table styles, but only AutoCAD on Windows allows you to create and modify table styles using the Table Style dialog box (shown in Figure 2.13). The Table Style dialog box is displayed with the `tablestyle` command. Once a table is created, you can override the properties of a table and its cells using the Properties palette (Windows) or Properties Inspector (Mac OS).

You can create a table style with these steps:

1. On the Annotate tab, Tables panel, click the panel-launcher button located to the right of the Tables panel's title (or at the command prompt, enter **`tablestyle`** and press Enter).
2. In the Table Style dialog box (see Figure 2.13), click New. Enter a name for the new table style and, optionally, select a style to start with. Click Continue.
3. In the New Table Style dialog box, select a table direction in the General section.
4. *Optionally*, click Create A New Cell Style in the Cell Styles section. In the Create New Cell Style dialog box, enter a name for the new cell style and, optionally, select a style to start with. Click Continue.
5. Select a cell style to edit from the Cell Styles drop-down list in the Cell Styles section.
6. Edit the properties for the cell style on the General, Text, and Borders tabs.
7. Click OK to return to the Table Style dialog box.

8. Select the table style you want to make current and click Set Current.
9. Click Close.

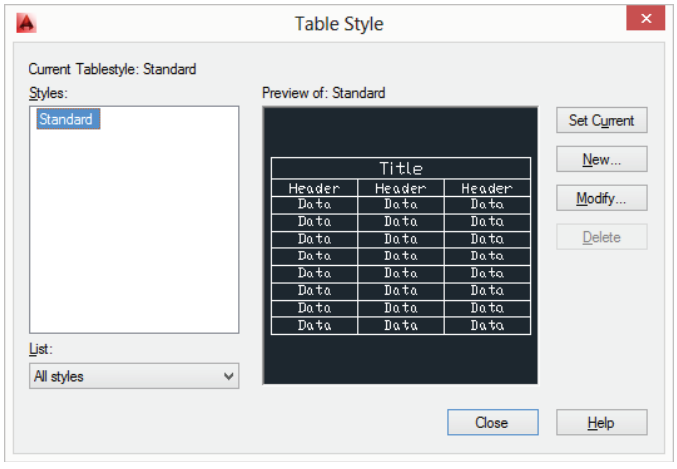


FIGURE 2.13: Updating the properties of a table style

You can update a table style with the Table Style dialog box by selecting the style you want to edit from the Table Styles list and clicking Modify. Edit the properties of the style as needed, and then click OK to save the changes and exit the dialog box.

Multileader Styles

Multileaders allow you to place text, blocks, or tolerances with leader lines that point to features in your drawing. You use multileader styles to control the creation and appearance of multileader objects. A multileader style specifies the formatting of the leader line (segment types, colors, and arrowhead style), leader structure (constraints, landing, and scale), and the content that should be displayed at the end of the leader landing. Multileader styles are created and modified using the Multileader Style Manager, which is displayed with the `mleaderstyle` command.

You can create a multileader style on Windows or Mac OS by taking these steps:

1. Do one of the following to display the Multileader Style Manager (see Figure 2.14):
   - On the ribbon's Annotate tab ➢ Leaders panel, click the panel-launcher button located to the right of the Leaders panel's title (Windows).
   - Click Format menu ➢ Multileader Style (Mac OS).
   - At the command prompt, enter **`mleaderstyle`** and press Enter (Windows and Mac OS).
2. In the Multileader Style Manager, click New (Windows) or the + (plus) button (Mac OS).
3. In the Create New Multileader Style dialog box, enter a name for the new style.
4. *Optionally*, select an existing style to start with and check Annotative if you want to create an annotative style. I discuss annotative styles in the next section, "Annotative Styles and Annotation Scaling."

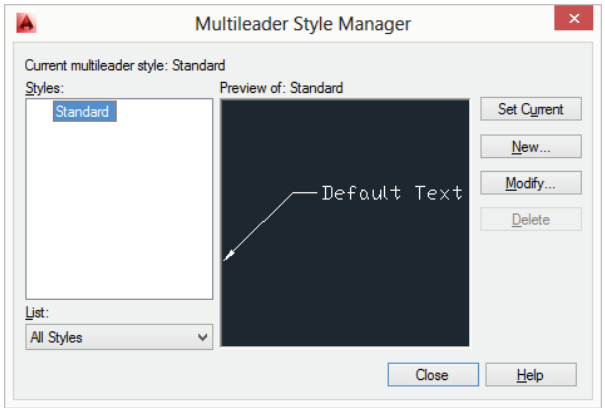


FIGURE 2.14: Modifying a multileader style

5. Click Continue. The Modify Multileader Style dialog box is displayed.
6. In the Modify Multileader Style dialog box, start on the Leader Format tab and change the settings as needed.
7. Continue making changes on each tab in the dialog box.
8. Click OK to return to the Multileader Style Manager.
9. Right-click the dimension style you want to make current and click Set Current. When no objects are selected in AutoCAD on Windows, you can also set a multileader style as current from the Multileader Style drop-down list on the Annotate tab ➢ Leaders panel on the ribbon. When no objects are selected in AutoCAD for Mac, you can use the Multileader Style drop-down list in the Annotation section of the Properties Inspector.
10. Click Close.

You can edit a multileader style by doing one of the following when the Multileader Style Manager is displayed:

- Select a multileader style from the Styles list box and click Modify (Windows).
- Select a multileader style from the Styles list box, click the Action button (the gear icon), and then click Modify (Mac OS).

Annotative Styles and Annotation Scaling

As I mentioned earlier, in the "Text Heights" section, calculating and displaying text at the correct height in a drawing can be a bit of a challenge when working with one or even several drawing scales. It can even be more time-consuming if you decide to change the scale at which your drawing should be plotted.

These problems can be solved through the use of annotative styles and annotation scaling. When you use annotative styles and annotation scaling, you specify the final height for your text objects and AutoCAD does the calculations for you based on the scale assigned to a viewport or when a drawing is plotted. For example, you set a text style to be annotative and then

specify a paper height (the size the text should appear on the drawing when plotted). Specifying a text height of 1/8″ means that the text will appear as 1/8″ on the sheet of paper as long as you have assigned the correct annotation scales to the text object. Entering 1/8″ is so much easier than calculating a text height of (1/8) × 96 = 12, so your text is printed as 1/8″ high when your viewport is set to a scale of 1/8″ = 1′-0″.

Okay, there are a few additional steps that need to happen, but it is much easier to let AutoCAD manage the display of annotation objects based on the current annotation scale than to create a large number of annotation objects at different sizes on different layers. The following styles can be set as annotative, which results in the creation of objects that are annotative and react to the current annotation scale in a drawing:

- Dimension styles
- Multileader styles
- Text styles

The following objects can be created using annotative styles or by enabling the Annotative property using the Properties palette (Windows) or Properties Inspector (Mac OS):

- Attribute definitions
- Block definitions/references
- Dimensions
- Geometric tolerances
- Hatch objects
- Multileaders
- Single-line and multiline text

Creating an Annotative Style

You can use the following to create an annotative text, dimension, or multileader style:

Text Style Create a text style as explained earlier, in the "Creating and Modifying Text Styles" section. Then, before saving the new style, in the Text Style dialog box check Annotative in the Size section in AutoCAD on Windows or click the Annotative toggle in AutoCAD for Mac. Enter the final height for the text in the Paper Text Height text box, and optionally click the Match Text Orientation To Layout check box (Windows) or toggle (Mac OS). Now the size of the text will be scaled up or down based on the annotation scales assigned to the single-line or multiline text objects that are created with that style current.

Dimension Style Create a dimension style as explained earlier, in the "Dimension Styles" section. Then, before saving the new style, in the New/Modify Dimension Style dialog box select the Fit tab and check Annotative in the Scale For Dimension Features section. Now the size, text height, and other distances that affect the appearance of dimension objects are scaled up or down based on the annotation scales assigned to the dimension objects that are created with that style current.

Multileader Style Create a multileader style as explained earlier, in the "Multileader Styles" section. Then, before saving the new style, in the Modify Multileader Style dialog box select the Leader Structure tab and check Annotative in the Scale section. Now the size, text height, and landing gap values on the Leader Format and Content tabs are scaled up or down based on the annotation scales assigned to the multileader objects that are created with that style current.

I explain creating annotative blocks in Chapter 3, "Building the Real World One Block at a Time."

Managing Annotation Scales for an Object

After an annotation object is created with an annotative style current or the object's Annotative property is enabled with the Properties palette (Windows) or Properties Inspector (Mac OS), you need to assign the annotation object one or more annotative scales. Annotative scales can be assigned to an annotation object using the `objectscale` command, or with the Annotative Scale property in the Properties palette (Windows) or Properties Inspector (Mac OS) when the object is selected.

Each annotative scale assigned to an annotation object creates an additional representation of the object; this allows you to control the placement of each representation independently while managing the content through a single object. Which representation of an annotation object should be displayed is determined by the current annotation scale of a viewport or when the drawing is plotted. For example, you can assign the annotative scales of 1/4″ = 1′-0″ and 1/8″ = 1′-0″ to your annotation objects, and if a viewport is set to the scale of 3/16″ = 1′-0″ none of the annotation objects are displayed. This is because your annotation objects were only assigned the scales of 1/4″ = 1′-0″ and 1/8″ = 1′-0″.

The current annotation scale for a viewport can be specified using the Annotation Scale drop-down list on the status bar. This value determines which annotation objects are displayed in the drawing based on their assigned annotative scales. You can add annotative scales automatically to all annotation objects when switching the value of the Annotation Scale drop-down list by enabling Automatically Add Scales To Annotative Objects on the status bar. If you are working in a drawing that has annotative objects with multiple annotative scales assigned to them, you might want to enable the Annotation Visibility option on the status bar as well so that you can adjust the placement of each representation of your annotation objects without switching between annotation scales.

Defining and Managing Other Nongraphical Objects

All of the drawings you create should contain a standard set of layers and annotation styles to ensure they have a consistent appearance from one drawing to the next. There are other nongraphical objects that you should consider standards for based on the types or sizes of the drawings that your company creates. Just as I recommended placing the layers and annotation styles that you frequently use in your drawing templates, you might want to do the same for some of the nongraphical objects mentioned in this section, especially if you work on 3D models.

Because these nongraphical objects are less frequently used, I mention them here only briefly; you can find more information by using the AutoCAD Help system. Most of these nongraphical

objects are supported on both Windows and Mac OS, but there are some limitations and I note some of those next.

Multiline Styles Multiline objects are used to create two or more parallel lines that might represent roadway, sidewalk, or utility offsets from a road. These offsets are defined as part of a multiline style. Although multiline styles are supported on both Windows and Mac OS, only the Windows version supports the `mlstyle` command, which allows you to create multiline styles directly from inside AutoCAD. On Windows and Mac OS, you can define multiline styles outside of AutoCAD by editing the `acad.mln` file with an ASCII text editor such as Notepad (Windows) or TextEdit (Mac OS).

View Styles View detail and section styles are used to control the appearance of associated detail and section views that are generated from 2D drawings created with AutoCAD and 3D models created with Autodesk Inventor®. You use the `viewdetailstyle` and `viewsectionstyle` commands to create and modify view detail and section styles. (This option is not supported on Mac OS.)

Views Panning and zooming in large drawings can become inefficient after a while, especially if you find yourself moving between the same areas of a drawing over and over again. Named views allow you to define a rectangular area of a drawing and give it a name. You can quickly return to that view by selecting its name from the Viewport Controls displayed in the upper-left corner of the drawing window. Named views are much more commonly used in 3D than 2D drawings to control the background of the current viewport. You use the `view` command to create named views. The Windows version of the command displays a dialog box, but the Mac OS version of the command is command-line–based only. Both versions of the command share the same options, with the exception of the Background and Shot Properties options, which are available on Windows only.

User Coordinate Systems (UCSs) All drawings contain a *world coordinate system* (WCS), which works well when you are drawing from a plan view. A user coordinate system (UCS) allows you to change the orientation of the working plane (X,Y) in addition to the Z-axis or placement of a drawing's origin. If you are drawing objects that are very far from the origin (0,0) of a drawing in 2D, adding ordinate dimensions, or using pattern fills, you might want to define a new origin to make entering coordinate values or controlling the pattern placement for a hatch object easier. UCSs are much more commonly associated with 3D modeling, though, because they allow you to align the UCS with the face of an object or change the direction of the Z-axis and the current working plane so that you can draw 2D objects at different angles in 3D space and then extrude them. The `ucs` and `ucsman` commands are used to create and manage named UCSs.

Model Space Viewports You can divide model space (using the Model tab) into more than one viewport and then save it as a viewport configuration. This approach is useful if you are working on a 3D model because it allows you to start a command in one viewport and switch to another viewport that has a different view to finish the command. It can be helpful when you are working on large drawings as well. You use the `vports` command to create named viewport configurations or restore the original single-viewport configuration.

Visual Styles Looking at 3D models in wireframe can make it hard to tell what is going on at times, and rendering your model can take some time based on the lighting, materials, and other rendering settings that are applied. Visual styles allow you to control a number

of settings that affect the appearance of 3D objects. They can be used to show only the visible edges from the current viewpoint, apply colors of shading, and even make 3D objects appear semitransparent to see the objects beyond. Visual styles can be created and modified in AutoCAD on both Windows and Mac OS, but the process is slightly different. In AutoCAD on Windows, you use the `visualstyle` command to display the Visual Styles Manager, but in AutoCAD for Mac you use a combination of the Visual Style section on the Properties Inspector (`properties` command) when no objects are selected and the `vssave` command to save the changes to a visual style.

Materials Materials help bring your 3D models to life by applying real-world materials to your virtual design. Using materials makes your models appear more alive and helps communicate what the final item or building might look like once manufactured or constructed. Materials can be assigned patterns, textures, colors, and other properties to represent tiles, bricks, glass, plastics, and much more. Both the Windows and Mac OS version of AutoCAD allow you to apply materials to 3D objects, but only AutoCAD on Windows lets you define and modify materials. You use the `materials` command when you want to work with materials.

Render Presets Rendering can be a complex process because of the number of settings that can be controlled. Using render presets makes this process much easier because it allows you to specify the rendering settings you commonly use and then save them as a custom render preset that you can use to render a 3D model to an image or when plotting. You can use render presets when rendering in AutoCAD on Windows or Mac OS, but render presets can be created only in AutoCAD on Windows with the `renderpreset` command.

Removing Unused Nongraphical Objects

Nongraphical objects cannot be simply removed from a drawing with the `erase` command, and only nongraphical objects that are not being referenced or used can be removed from a drawing. The removal of nongraphical objects is handled with the `purge` command. In addition to removing nongraphical objects from a drawing, the `purge` command can be used to remove registered application IDs created by third-party programs, non-zero–length lines, and empty text objects. It is good practice to remove the nongraphical objects you no longer need or are not using.

Follow these steps to remove all unused nongraphical objects from the current drawing in AutoCAD for Windows:

1. Click the Application button ➢ Drawing Utilities ➢ Purge (or at the command prompt, type **`purge`** and press Enter).
2. In the Purge dialog box, clear Confirm Each Item To Be Purged and check Purge Nested Items. The Confirm Each Item To Be Purged option, when checked, displays a message box that prompts you to verify the removal of each unused nongraphical object, and the Purge Nested Items option checks for additional nongraphical objects after all the currently identified unused objects have been removed.
3. Click Purge All. Doing so removes all unused nongraphical objects from the drawing.
4. Click Close.

Follow these steps to remove all unused nongraphical objects from the current drawing using the command-line version of the `purge` command available in AutoCAD for Windows or Mac OS:

1. At the command prompt, type **`-purge`** and press Enter.
2. At the `Enter type of unused objects to purge [Blocks/DEtailviewstyles /Dimstyles/Groups/LAyers/LTypes/MAterials/MUltileaderstyles/Plotstyles /SHapes/textSTyles/Mlinestyles/SEctionviewstyles/Tablestyles/Visualstyles /Regapps/Zero-length geometry/Empty text objects/All]:` prompt, type **`all`** and press Enter.

NOTE The command-line version of the `purge` command does not purge nested nongraphical objects; that can be done only by running the `purge` command multiple times.

Chapter 3

Building the Real World One Block at a Time

Drawings are made of geometry that is arranged in specific ways to represent a part to be manufactured, a building to be constructed, or even the land on which a new highway might be built. No matter what type of drawings you create, your drawings most likely consist of many object groupings that represent objects in the physical world.

Instead of dedicating time to re-create the geometry each time, you can create the objects once and store them as a *block definition*. Block definitions are another type of nongraphical object that can be in a drawing, much like text and dimension styles.

In addition to geometric objects, block definitions can contain attribute definitions, which allow you to embed information into a block. The information stored in attributes can be extracted to an external file or added to a table in a drawing. When the `insert` command or one of the programming languages is used, the Autodesk® AutoCAD® program creates a block reference object that specifies where and how the objects from the block definition should be drawn onscreen or plotted.

Defining and Editing a Block Definition

Block definitions must be defined in a drawing before you can create (or, as commonly known, insert) a *block reference*. A block reference object specifies where and how the objects from a block definition should be drawn onscreen or plotted. You typically create block definitions by first adding the geometry for your block in model space and then using the `block` command to define which objects will make up the block definition. AutoCAD also offers a special environment called the Block Editor (`bedit` command) for working with block definitions. Once a block definition is defined, you can use the `insert` command to create a block reference of a block definition.

Defining a Block Definition

You do not have to learn a lot of new skills before creating a block definition since the process leverages many skills that you should already have, such as creating geometry and using nongraphical objects like layers and text styles. However, when defining a block you must consider the following:

Name The name of a block definition is used to differentiate one from another; each block definition in a drawing must have a unique name. Drawings can easily contain hundreds

and even thousands of block definitions. When naming block definitions, use descriptive names, but don't make them so long that the user must carefully read a sentence-long name. I discussed naming blocks and other nongraphical objects in Chapter 2, "Working with Nongraphical Objects."

Base (or Insertion) Point The base point for a block definition is similar to a drawing's origin. This point is used as a block reference's insertion point, the same point that is used to drag a preview of the block onscreen with the `insert` command. You typically specify the base point of a block on one of the objects in the block definition, such as an endpoint or intersection of two objects.

Objects The objects that make up the block definition determine how the block references inserted into a drawing will look and behave. Objects within a drawing include geometry objects such as lines, circles, arcs, and even other blocks. Blocks can also include special objects known as *attribute definitions*, which allow you to add to a block information such as a project name or part number. I discuss attributes in the section "Embedding Information in a Block Definition with Attributes" later in this chapter.

The following steps can be used to create a block definition from objects in your current drawing:

1. Draw the geometry that you want to use when defining the block definition in model space or paper space. The object properties used to create the geometry determine how those objects will look when the block definition is used to insert a block reference. I explain the impact of object property values in the sidebar "Assigning Object Properties By Layer, By Block, or By Object."
2. Do one of the following:
   - On the ribbon, click Insert tab ➢ Block Definition panel ➢ Create Block drop-down menu ➢ Create Block (Windows).
   - Click Draw menu ➢ Block ➢ Make (Mac OS).
   - At the command prompt, enter **`block`** and press Enter (Windows and Mac OS).
3. When the Block Definition (Windows) or Define Block (Mac OS) dialog box opens, enter a name for the block definition in the Name text box (see Figure 3.1). You can select the name of an existing block definition from the Name or Blocks drop-down list to redefine the geometry and base point of a block definition. I discuss redefining blocks in the section "Editing a Block Definition" later in this chapter.
4. In the Base Point section, click Pick Point and specify a coordinate point in the drawing, or enter coordinate values directly into the text boxes. I recommend using object snaps to specify a point on one of the geometric objects that is part of the block.
5. In the Objects/Source Objects section, click Select Objects and select the objects that will make up the block definition. Press Enter to complete object selection and return to the dialog box.

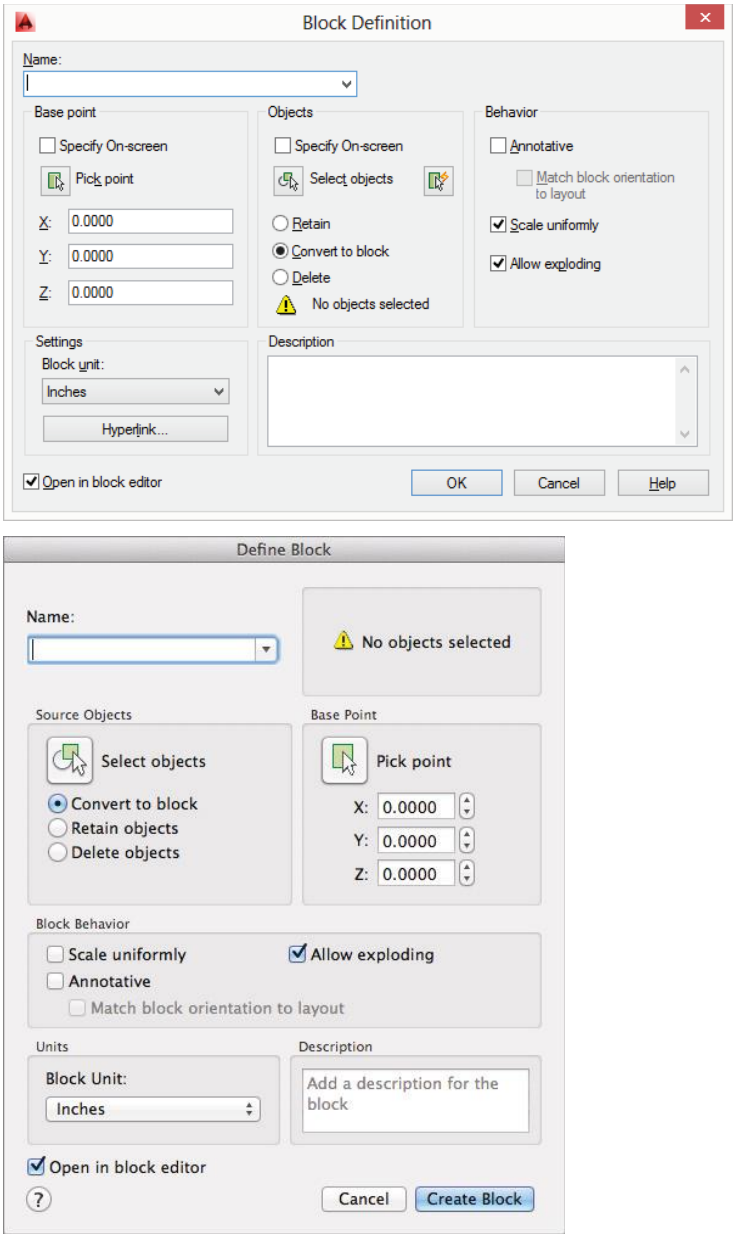


FIGURE 3.1
Defining a block

6. In the Objects/Source Objects section, choose one of the following:
   - Retain/Retain Objects: When you click Retain/Retain Objects, the selected objects are not removed from the drawing when the block definition is created, nor are they replaced with a block reference.
   - Convert To Block: If you select Convert To Block, the selected objects are removed from the drawing and replaced by a block reference based on the new block definition.
   - Delete/Delete Objects: When Delete/Delete Objects is enabled, the selected objects are removed from the drawing and not replaced with a block reference based on the new block definition.
7. In the Behavior/Block Behavior section, choose any of the following options to control the behavior of the block references that are based on the block definition:
   - Annotative: When Annotative is enabled, the block definition is designated as being *annotative*. Annotative block references are scaled based on the current annotation scale of the current viewport or model space (Model tab). I discussed annotative objects and annotation scaling in Chapter 2.
   - Match Block Orientation To Layout: When this option is enabled and the block reference is displayed in a viewport on a named layout, the orientation of the block reference changes to match that of the named layout. The Annotative check box needs to be enabled to use this option.
   - Scale Uniformly: When Scale Uniformly is enabled, the block reference is forced to be scaled equally across all axes. If the Annotative check box is enabled, this option is enabled and cannot be disabled.
   - Allow Exploding: When Allow Exploding is enabled, it allows the block reference that is inserted into a drawing to be exploded. Exploding a block reference removes it from the drawing and it is replaced with a copy of the geometry from the block definition. Attributes values are lost when you use the `explode` command, but they can be preserved if you are working on Windows by using the `burst` Express Tool.
8. In the Settings/Units section, choose one of these options:
   - Block Unit: Specifies the units in which the block should be assigned. The unit value chosen affects the scale at which the block is inserted into a drawing. For example, if the block is defined using inches and it is inserted into a drawing that is set to use millimeters, the block is scaled automatically so that it is displayed at the correct size. The drawing units used to scale blocks when they are inserted is determined by the `insunits` system variable. The `insunitsdefsource` and `insunitsdefsource` system variables also impact how blocks are scaled when inserted into a drawing when they are defined as Unitless.
   - Hyperlink (Windows only): Allows you to add a hyperlink to a block. When a block is inserted with a hyperlink, you can hold the Ctrl key and click the block to open the associated file, drawing view, or URL.
9. Optionally, in the Description section, enter a description that can provide some information about the block.

10. Optionally, click the Open In Block Editor check box to open the new block definition in the Block Editor. I discuss the Block Editor in the section "Using the Block Editor" later in this chapter.
11. Click OK (Windows) or Create Block (Mac OS).

Assigning Object Properties By Layer, By Block, or By Object

Layers and object properties play an important role in how the objects of a block definition appear when inserted as a block reference into model space or paper space. The general properties of an object in a block can be controlled using one of the following methods:

By Layer An object's properties are controlled by the layer the object is on. For an object to inherit a property value from a layer, the property must be set to `ByLayer`.

By Object An object's properties values can be set directly, and they are not affected by the layer the object is on or the properties of the inserted block reference.

By Block An object's properties are controlled by the block reference that is inserted into model space or paper space. For an object to inherit a property value from a block reference, the object's property must be set to `ByBlock`.

Editing a Block Definition

A block definition can be edited (or redefined, as you might hear some say), but the editing process is not as simple as just selecting an object and making changes to it. Since a block definition and its objects do not exist in model space or paper space, you need to do one of the following:

- Create a block reference with the `insert` command in either model space or paper space, and then explode it with the `explode` command. Once it's exploded, you can modify the objects that were created by exploding the block reference and then use the `block` command to define the block definition again with the same name.
- Open a block reference in the drawing area with the In-Place Reference Editor (`refedit` command). This allows you to add and modify the objects of the block definition and use the other objects in your drawing as reference objects.
- Open a block definition in the Block Editor (`bedit` command). This provides a separate environment for creating and modifying geometry and custom dynamic properties, and for testing your block definition.
- If the block definition was originally stored as a separate drawing, you can open that drawing, make changes to it, and then reinsert it into your drawing with the same name using the Insert/Insert Block dialog box (`insert` command). You can export a block to a separate drawing by using the Write Block dialog box (`wblock` command).

NOTE When you edit or redefine the geometry of a block definition, all block references that use the block definition in the current drawing are updated to reflect the changes. If you want

only a few block references to be different, you can use the Block Editor's `bsaveas` command to create a new block definition from an existing one, and then replace the block references that should be different.

Follow these steps to edit the geometry of a block definition without the In-Place Reference Editor or Block Editor:

1. Do one of the following:
   - On the ribbon, click Insert tab ➢ Block panel ➢ Insert (Windows).
   - Click Insert menu ➢ Block (Mac OS).
   - At the command prompt, enter **`insert`** and press Enter (Windows and Mac OS).
2. When the Insert/Insert Block dialog box opens, select a block definition from the Name Or Blocks drop-down list. Check Explode/Explode Block, and then click OK or Insert. Make sure the rotation of the block reference is set to 0 and the scale of the block reference is set to 1 and scaled uniformly. For more information on the Insert/Insert Block dialog box, see the section "Inserting or Replacing a Block Reference" later in this chapter.
3. In the drawing area, modify the objects that were added as a result of exploding the block reference.
4. Use the `block` command to create a block definition with the same name. Make sure to select the name of the block definition you are updating from the Name Or Blocks drop-down list. Make sure that you assign the same base point as before, because the new value will impact all block references that use the block definition you are updating. Block units is also important for all future block references being inserted.

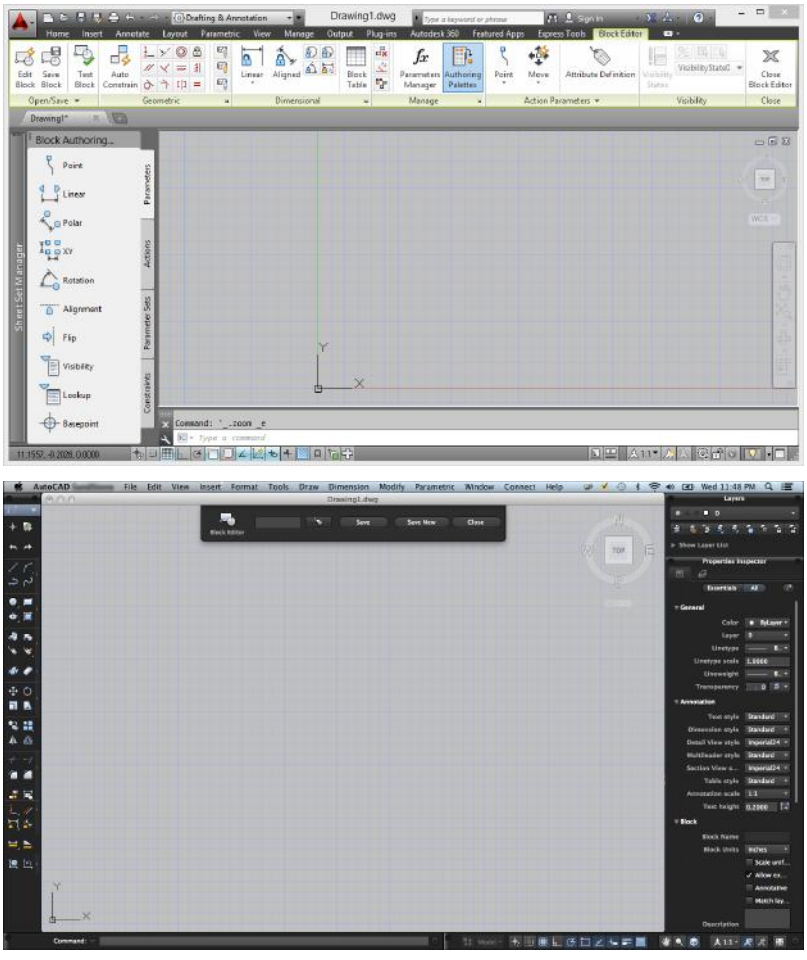
Using the Block Editor

The Block Editor (shown in Figure 3.2) allows you to create and modify the block definitions in the current drawing. You have a choice to open the Block Editor when you create a new block definition with the Block Definition (Windows) or Define Block (Mac OS) dialog box, or when you want to modify an existing block definition with the `bedit` command. The Block Editor makes the process of updating a block definition much easier than having to insert a block reference, then explode it, and finally define a block definition from the modified objects.

When a block definition is open for editing in the Block Editor, you can perform the following tasks:

- Add new, remove, and modify existing geometry and its properties
- Manage attribute definitions
- Add parameters and actions (Windows only)
- Work with lookup tables and visibility states (Windows only)

While the Block Editor is open, there are some commands that you can't use. Some of those are `block`, `refedit`, and `plot`. When a command can't be used, the message `** <name> command not allowed in block editor. **` is displayed in the command-line history.



Components of the Block Editor

The Block Editor is similar to the normal drawing window, but it offers several additional tools that are specific to working with objects in a block definition. When the Block Editor is active, the Block Editor tab (see Figure 3.3, top) is displayed on the ribbon in AutoCAD on Windows and the Block Editor visor (see Figure 3.3, bottom) is displayed in AutoCAD for Mac.

Figure 3.2: Block editing environment

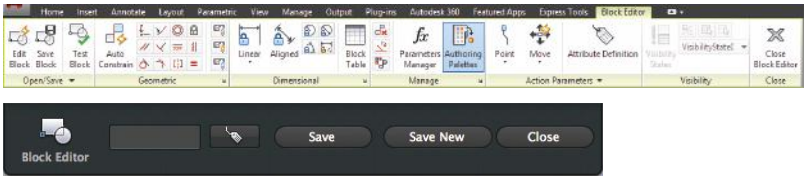


FIGURE 3.3 Tools available for modifying the objects and behavior of a block definition

The following describes the tools on the Block Editor tab in AutoCAD on Windows:

Open/Save These tools allow you to open a block definition in the Block Editor (`bedit` command), save changes to the open block definition (`bsave` command), create a new block definition (`bsaveas` command), or test a block definition (`btestblock` command).

Geometric These tools allow you to create and modify geometric constraints and control the display of the constraint bars for the geometric constraints placed in the block definition (`constraintbar` command). Geometric constraints can be placed automatically with the AutoConstrain tool (`autoconstrain` command) as you create new geometry or manually as needed (`geomconstraint` command). The settings used for the AutoConstrain tool can be adjusted with the Constraint Settings dialog box (`constraintsettings` command).

Dimensional These tools allow you to create and modify constraint parameters. Constraint parameters can be applied to selected objects, or you can convert a dimensional constraint to a parameter (`bcparamter` command). The display of dimensional constraints can be controlled with the Constraint Settings dialog box (`constraintsettings` command). You can also create a Block Properties table (`btable` command) that can be used to define variations of a block reference.

Manage These tools are used to delete (`delconstraint` command) and control the display of constraints (`bconstatusmode` system variable), as well as convert geometry to construction objects (`bconstruction` command). You can also display the Parameters Manager (`parameters` command) and the Block Authoring palette (`bauthorpalette` command), in addition to controlling the display settings of the Block Editor (`bsettings` command).

Action Parameters The tools here allow you to create actions (`bactiontool` command) and parameters (`bparameter` command) to define a dynamic block. You can use the Attribute Definition button to display the Attribute Definition dialog box to add an attribute definition to the block definition (`attdef` command). I discuss actions and parameters in the section "Adding Dynamic Properties to a Block Definition" and attributes in the section "Embedding Information in a Block Definition with Attributes" later in this chapter.

Visibility These tools allow you to create and manage visibility states (`bvstate` command), and control the visibility of the geometry of a visibility state (the `bvhide` and `bvshow` commands and the `bvmode` system variable). I discuss visibility states in the section "Adding Dynamic Properties to a Block Definition" later in the chapter.

Close The Close Block Editor button closes the Block Editor and prompts you to save or discard the changes made to the block definition (`bclose` command).

The following describes the tools on the Block Editor visor in AutoCAD on Mac OS:

Block Definition Name This tool displays the name of the block definition open in the Block Editor.

Attribute Definition The Attribute Definition tool displays the Attribute Definition dialog box and allows you to add a new attribute definition to the block definition (`attdef` command). I discuss attributes in detail in the section "Embedding Information in a Block Definition with Attributes" later in this chapter.

Save Click this button to save the changes made to the block definition (`bsave` command).

Save New Click Save New to open the Save Block Definition As dialog box and create a new block definition based on the one open in the Block Editor (`bsaveas` command). Enter a new name for the block and click Save Block. If you want to export the new block definition to a new drawing file, check Save Block As Drawing before clicking Save Block.

Close This button closes the Block Editor and prompts you to save or discard the changes made to the block definition (`bclose` command).

Creating or Editing a Block Definition

The Block Editor simplifies the creation and editing process of a block definition, but it does not allow you to use other geometry in your drawing as reference objects. If you are using AutoCAD on Windows, the Block Editor is the only way to add actions and parameters to a block definition. I discuss actions and parameters for block definitions in the section "Adding Dynamic Properties to a Block Definition" later in this chapter.

You can use the following steps to create a new block definition or edit an existing one with the Block Editor:

1. Do one of the following:
   - On the ribbon, click Insert tab ➢ Block Definition panel ➢ Block Editor (Windows).
   - Click Tools menu ➢ Block Editor (Mac OS).
   - At the command prompt, enter **`bedit`** and press Enter (Windows and Mac OS).
2. In the Edit Block Definition dialog box, do one of the following:
   - Create a new block definition: Enter a name in the Block To Create Or Edit text box in the upper-left corner.
   - Edit an existing block definition: Select a name from the Block Definition list or click <Current Drawing> to edit the objects in model space in the Block Editor.
3. Click OK or Edit Block to open the block definition in the Block Editor.
4. In the Block Editor, add or modify the geometry of the block definition as needed.
5. Do one of the following:
   - On the ribbon, click Block Editor tab ➢ Close panel ➢ Close Block Editor (Windows).
   - On the Block Editor visor, click Close (Mac OS).
   - At the command prompt, enter **`bclose`** and press Enter (Windows and Mac OS).
6. In the Block - Changes Not Saved message box, click Save The Changes To *<definition name>*.

NOTE The `blockeditlock` system variable controls whether you can use the Block Editor to edit block definitions. A value of `0` disables the use of the Block Editor.

Using and Managing Block Definitions

After a block definition has been created in a drawing, you can place a block reference for that definition in model space or paper space with the `insert` command. In addition to inserting block references, you can replace a block reference with a different block definition, rename and purge unused block definitions, import a drawing file as a block definition, or export a block definition to a drawing file.

NOTE The `insert` command does not create a copy of the objects contained in the block definition before placing them in model space or paper space. This can be seen by making a change to a block definition; all block references that point to a particular block definition are updated with the changes made to the block definition.

Inserting or Replacing a Block Reference

You can insert a drawing file or block definition in the current drawing as a block reference using the Insert (Windows) or Insert Block (Mac OS) dialog box, which is displayed with the `insert` command. In addition to the `insert` command, you can insert block references using the following:

- The AutoCAD DesignCenter palette on Windows (`adcenter` command)
- The Content Explorer palette on Windows (`contentexplorer` command)
- The Tool Palettes window on Windows (`toolpalettes` command)
- Drag and drop, which you can use to move a drawing from Windows Explorer or File Explorer onto a drawing window when using AutoCAD on Windows
- The Content window on Mac OS (`content` command)
- The `-insert` command, which you can use to insert commands from the command prompt using scripts, menu or action macros, and even AutoLISP® programming language

Follow these steps to insert a block reference into a drawing with the `insert` command:

1. Do one of the following:
   - On the ribbon, click Insert tab ➢ Block panel ➢ Insert (Windows).
   - Click Insert menu ➢ Block (Mac OS).
   - At the command prompt, enter **`insert`** and press Enter (Windows and Mac OS).
2. In the Insert or Insert Block dialog box (see Figure 3.4), do one of the following:
   - Insert a drawing file: Click Browse. When the Select Drawing File dialog box opens, browse to and select the drawing file that you want to insert. Click Open.
   - Insert a block definition: From the Name (Windows) or Blocks (Mac OS) drop-down list, select the block definition you want to insert.

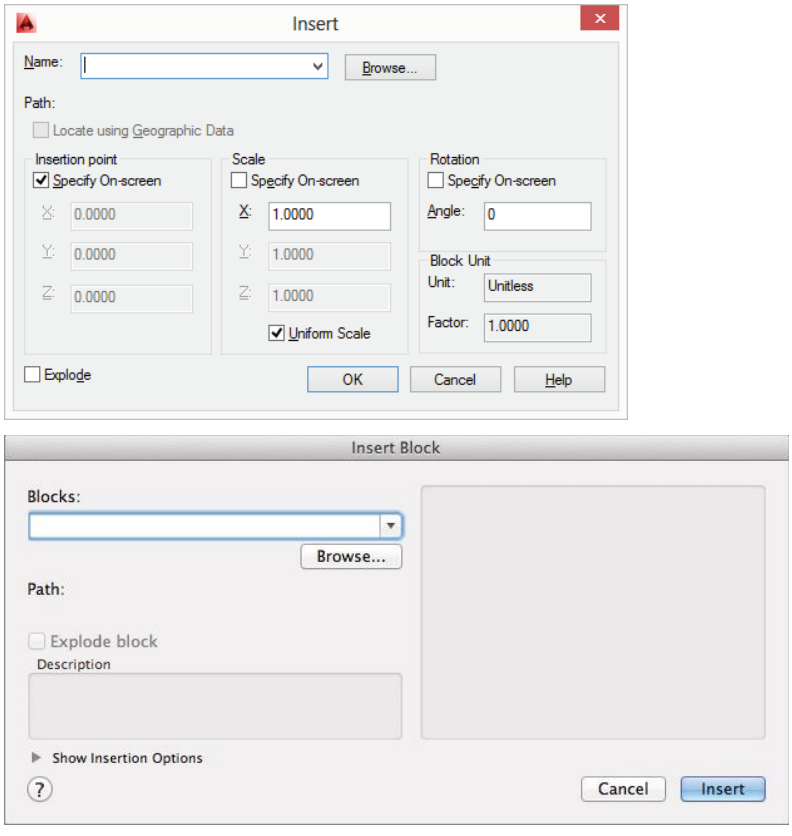


FIGURE 3.4
Inserting a block definition

3. If you are using AutoCAD for Mac, click the disclosure triangle to the left of Show Insertion Options.
4. In the Insertion Point, Scale, or Rotation section, enter the values you want to use when inserting the block reference, or choose Specify On-Screen for the values you want to provide at the command prompt when inserting the block reference.
5. Optionally, enable any of the following:
   - Explode/Explode Block: Explodes the block reference into its individual objects once it is inserted into the drawing.
   - Uniform Scale/Scale Uniformly: Sets the Y and Z scale factors to be the same as the X scale factor
   - Locate Using Geographic Data (Windows only): Inserts the selected drawing and locates it based on the geographic data in the current drawing and in the drawing file you are inserting
6. Click OK (Windows) or Insert (Mac OS) to insert the block reference. Specify the insertion point, scale, or rotation in the drawing area based on the options you chose in step 4.

NOTE As a block reference is being inserted (dragged onscreen) into the drawing area, you can use the Properties palette in AutoCAD on Windows to control the block reference's properties, attribute values, and custom dynamic properties.

After a block reference is inserted, at times you will want to replace one or all block references with another block. You can use one of the following methods to replace one or all block references in a drawing:

Replace a Block Definition with Another You can replace a block definition with another by using either the `-insert` command or the Replace Block Express Tool (`blockreplace` command) in AutoCAD on Windows. For more information on the Replace Block Express Tool, look up "blockreplace" in the product's help.

The following example explains how to replace a block definition with the geometry from a drawing file using the `-insert` command:

1. At the command prompt, enter **`-insert`** and press Enter.
2. At the `Enter block name or [?]:` prompt, enter *`blockdef1=blockdef2`* and press Enter. Replace *`blockdef1`* with the name of the block definition you want to replace and *`blockdef2`* with the name of the block definition you want to replace it with.
3. At the `Block "<blockdef1>" already exists. Redefine it? [Yes/No] <N>:` prompt, enter **y** and press Enter.
4. At the `Specify insertion point:` prompt, press Esc to exit the `-insert` command.

NOTE You can export a block definition from a drawing to a drawing file using the Write Block dialog box (`wblock` command). I discuss the `wblock` command later in this chapter in the section "Importing or Exporting a Block Definition."

Replace an Individual Block Reference with Another The `-insert` and `blockreplace` commands allow you to replace a block definition with another, but not to replace a block reference with another block reference. You can use one of the available programming languages to create a basic routine that allows you to do that. The following AutoLISP code demonstrates a basic routine for swapping one block reference for another.

```
(defun c:ReplaceBlk ( / ed newBlk)
  (setq ed (entget (car (entsel "\nSelect block to replace: "))))
  (if (setq newBlk (getstring "\nEnter name of new block to use: "))
    (progn
      (setq ed (subst (cons 2 newBlk) (assoc 2 ed) ed ))
      (entmod ed)
    )
  )
)
```

Renaming and Purging Unused Block Definitions

Block definitions can be renamed and purged from a drawing just like other nongraphical objects stored in a drawing. As you'll recall, you can rename a block definition using the `rename`

command. I discussed renaming nongraphical objects in Chapter 2. Block definitions that are no longer needed in a drawing can be removed using the `purge` command. Before removing a block definition, you might want to consider using the `wblock` command to export the block definition to a drawing file just in case you need it later and it does not exist elsewhere. I discuss exporting a block definition to a drawing file in the next section, and you learned about purging nongraphical objects in Chapter 2.

Importing or Exporting a Block Definition

You can import a drawing file into a drawing as a block definition by using the Insert (Windows) or Insert Block (Mac OS) dialog box (`insert` command), or if you are using AutoCAD on Windows, you can import a block definition from another drawing using DesignCenter (`adCenter` command), Content Explorer (`contentexplorer` command), or the Tool Palettes window (`toolpalettes` command). In AutoCAD for Mac, you can use the `insert` command or Content palette (`content` command). After selecting a drawing file or block definition to insert with one of the previously mentioned commands, press Esc when prompted for an insertion point. The block definition is imported, but the block reference is not inserted into model space or paper space.

If you defined a block definition in a drawing, you can use the Write Block dialog box (`wblock` command) to save the block definition to a separate drawing file. The new drawing file can then be used to import a block definition into another drawing. In addition to exporting a block definition, you can save all the objects in model space or select which objects to save to a drawing file when using the `wblock` command.

Embedding Information in a Block Definition with Attributes

Block definitions can contain attributes that allow you to add custom information to a block reference when it is inserted into a drawing. Using attributes allows you to use a single block definition but represent several items that might be the same size and a different color or style. A common use for attributes is to store project, date, and revision information in a title block, or a model number and description with a window or bolt block. After you insert a block reference with attributes, you can extract the values stored in the attributes to an external file or table object.

There are two types of attributes in a drawing. Those in a block definition are known as *attribute definitions*. The other type of attribute is known as an *attribute reference*, which can be found attached to a block reference that has been inserted into a drawing whose block definition contains attribute definitions.

Adding an Attribute Definition

You can create attribute definitions by using the Attribute Definition dialog box (`attdef` command). You can add an attribute definition to model space or paper space before you define your block definition by using the Block Definition (Windows) or Define Block (Mac OS) dialog box (`block` command), or after a block definition is created in the Block Editor (`bedit` command).

The following steps explain how to add an attribute definition, whether you are working in the drawing window or the Block Editor:

1. Do one of the following:
   - On the ribbon, click Insert tab ➢ Block Definition panel ➢ Define Attributes (Windows). When the Block Editor is active, on the ribbon, click Block Editor tab ➢ Action Parameters panel ➢ Attribute Definition.
   - Click Insert menu ➢ Block ➢ Define Attributes (Mac OS). When the Block Editor is active, on the Block Editor visor, click Attribute Definition.
   - At the command prompt, enter **`attdef`** and press Enter (Windows and Mac OS).
2. In the Attribute Definition dialog box (see Figure 3.5), in the Attribute section's Tag text box, enter a text string that doesn't contain any spaces. The Tag value is shown in the drawing and the Block Editor when modifying a block definition, and it is the internal identifier of the attribute within the block. You should use a different tag value for each attribute in a block definition so that you can accurately extract the attribute values you might be interested in later.
3. Optionally, in the Attribute section enter a text string in the Prompt and Default text boxes. The text string you enter in the Prompt text box is displayed after you insert a block reference with the `insert` command or while you are editing the attribute values of a block reference. The text string in the Default text box is the default value that an attribute is assigned when a block reference is inserted. You can also use the Insert Field button to add a field as the attribute definition's default value instead of a static text string.
4. In the Mode section, set one or more options that define how the attribute reference should behave when inserting the block reference. On Mac OS, click the disclosure triangle next to Show Advanced Options to see the Mode section. The following explains what each mode does:
   - Invisible: The attribute is invisible when a block reference is inserted into the drawing. All invisible attributes can be displayed by setting the `attmode` system variable to a value of 2.
   - Constant: The text string entered in the Value text box of the attribute is fixed and is the same for all block references that are inserted based on the block definition that contains this type of attribute.
   - Verify: You are prompted to verify the value of the attribute when the block reference is inserted.
   - Preset: The Default value is assigned to the attribute when the block reference is inserted, and you are not prompted to change the value. You can change the value using the `attedit` or `eattedit` command or the Properties palette (Windows) or Properties Inspector (Mac OS).
   - Lock Position: Restricts the attribute from being moved using grips. This mode is helpful if you want to control the placement of an attribute with parameters and actions in a dynamic block.

- Multiple Lines: The attribute supports multiple lines of text similar to a multiline text (MText) object, instead of the default single line of text. You can control whether multiline attributes can be created or which Text Formatting toolbar is displayed by using the `attmulti` and `attipe` system variables.

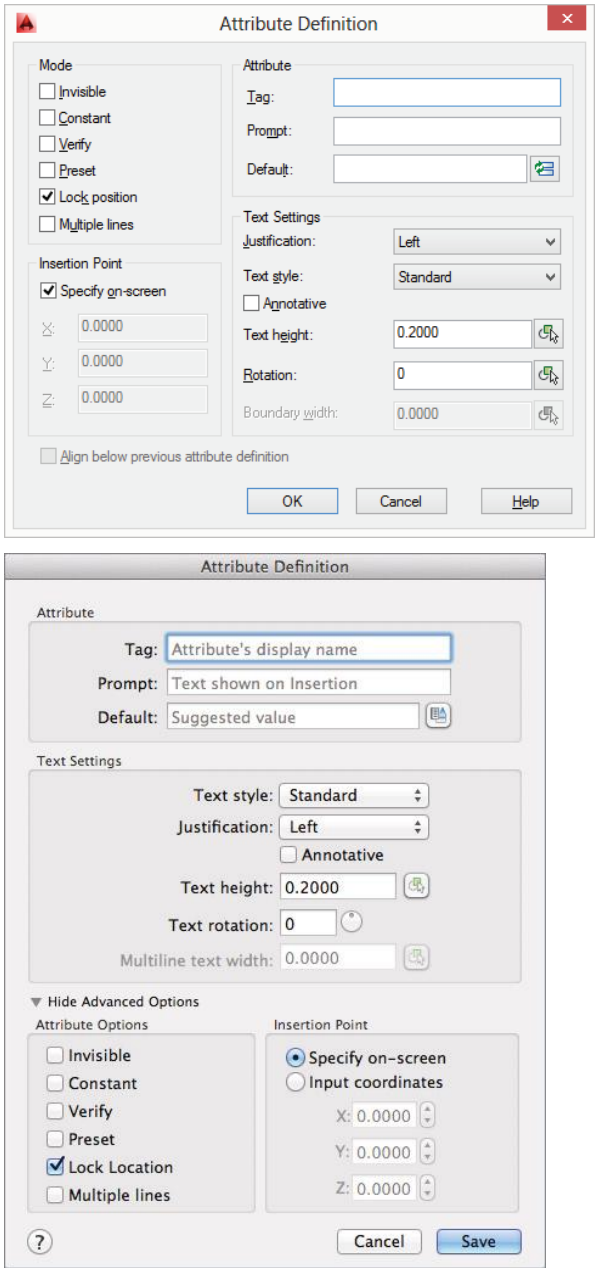


FIGURE 3.5 Defining an attribute definition

NOTE While AutoCAD remains open, the Mode options you choose are stored as a sum of different bitcode values in the `aflags` system variable. The value of `aflags` is used as the default option for the next attribute definition you create.

5. In the Insertion Point section, choose Specify On-Screen to specify a point in the drawing area after you create the attribute definition or enter a coordinate value. After a block reference with an attribute is inserted, you can use grips or the `-attedit` command to move an attribute as long as it is not set to the Invisible mode.
6. In the Text Settings section, change any of the following:
   - Justification: The Justification setting controls how the text string assigned to the Value property of an attribute appears in relation to the attribute's insertion point.
   - Text Style: This setting specifies the text style that should be used for the attribute. I discussed creating text styles in Chapter 2.
   - Annotative: This setting controls whether the attribute is annotative. If it's enabled and the block is annotative, the attribute matches the orientation of the block. I discussed annotative objects and styles in Chapter 2.
   - Text Height: This setting specifies the height at which the attribute should be created. The height of the attribute scales up or down when it and the block definition it is in are annotative.
   - Rotation/Text Rotation: Using this setting, you can specify the rotation angle at which the attribute appears when the block reference is inserted into a drawing. After a block reference with an attribute is inserted, you can use the `-attedit` command to rotate an attribute as long as it is not set to the Invisible mode.
   - Boundary Width/Multiline Text Width: When the Multiple Lines mode is enabled, you can specify the width of the boundary box for the multiline attribute object. Additional lines of text are created vertically based on the length of the text string assigned to the attribute's Value property.
7. Optionally in AutoCAD on Windows, click Align Below Previous Attribute Definition if you have created another attribute definition and want to align the new and previous attribute definitions vertically.
8. Click OK (Windows) or Save (Mac OS) to create the attribute definition. Specify the insertion point in the drawing area based on the option you chose in step 5.
9. Repeat steps 1–8 for each attribute definition you want to add to a block.
10. Create a block definition or save the changes in the Block Editor as I discussed in the sections "Defining and Editing a Block Definition" and "Creating or Editing a Block Definition" earlier in this chapter.

NOTE When you are selecting objects to define a block definition, the order in which you select your attribute definitions affects how AutoCAD prompts for each attribute value. I recommend selecting all your graphical objects first, and then selecting each attribute definition in the order you want its value to be prompted for. Prompting order can have an impact on your block's usage with scripts, menu macros, and other custom programs. You can also change the prompting order of the attributes in a block by using the Block Attribute Manager (`battman` command).

Inserting a Block Definition with Attributes

Inserting a block reference with attributes follows the same process as inserting a block reference without attributes, with one difference: you might need to accept the default values or enter a new value for the attributes being added after the block reference has been placed in a drawing. When you insert a block reference with attributes, the Edit Attributes dialog box (see Figure 3.6) is displayed and allows you to provide new values for the attribute references in the block reference. Attributes that have Constant mode enabled are not displayed in the dialog box.

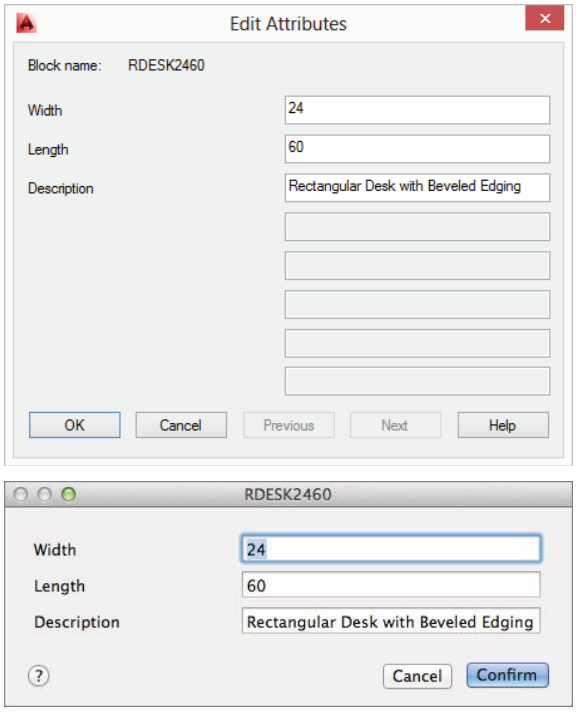


Figure 3.6
You can edit attribute values after inserting a block reference.

The display of the Edit Attributes dialog box and prompting for attribute values is controlled with the `attdia` and `attreq` system variables. The `attdia` system variable controls the display of the Edit Attributes dialog box. If the dialog box does not appear after you insert a block, you are instead prompted to enter attribute values at the command prompt. The `attreq` system variable enables or disables the prompt for attribute values after you insert a block reference.

Editing Attribute Values and Properties in a Block Reference

After a block reference with attributes has been inserted into a drawing, you can change the value assigned to an attribute using one of the following methods:

Edit Attributes Dialog Box The `attedit` or `ddedit` commands allow you to edit the values of the attributes in a block reference with the Edit Attributes dialog box.

Properties Palette or Properties Inspector You can modify an attribute's value by selecting a block reference and then displaying the Properties palette (Windows) or Properties Inspector (Mac OS). The attributes and their editable values are listed in the Attributes

section. You can display the Properties palette or Properties Inspector by using the `properties` command.

Enhanced Attribute Editor The values and properties of an attribute in a block reference can be modified using the Enhanced Attribute Editor (`eattedit` command). You can also double-click a block reference with attributes to open the Enhanced Attribute Editor.

`-attedit` Command If you plan to write scripts, action macros, or custom programs that edit the values and properties of the attributes in a block reference, you can use the `-attedit` command. This command allows you to perform a global find and replace on a text string in an attribute value and edit the properties of an attribute.

Redefining Block Definitions with Attribute Definitions

Just as it is not uncommon to make changes to the way a block looks over time, you might decide to update the attribute definitions in a block definition. There are a few things you need to consider when adding, modifying, or removing attribute definitions from a block definition. Although it is true that the geometry of a block definition is the same for each and every block reference that uses a specific block definition, the same is not true for attribute definitions and references unless all your attribute definitions are defined as Constant.

Some blocks based on the same block definition may not have all the attributes defined in that block definition. This situation often occurs when attributes in a block definition are updated but the block references themselves are not synchronized to include the same changes, or when a custom application has decided not to add an attribute reference to a block reference for some reason.

After you update the attributes of a block definition, you want to run the `attredef` or `attsync` command or use the Block Attribute Manager (`battman` command). These utilities synchronize the attribute definitions in a block definition with the attribute references in each of the block references that have been inserted in a drawing by adding and removing attributes as needed. Those attributes in the block reference that are contained in the block definition are not changed; their current values are retained.

The Block Attribute Manager (`battman` command) can also be helpful in updating the properties and prompting order of the attribute definitions in a block definition, as well as editing and removing attribute definitions. These three commands can also affect any formatting or property changes made to attributes in a block reference with the `attedit` and `eattedit` commands. If you need to just change the prompting order of the attributes within a block definition, you can use the `battorder` command while the block definition is open in the Block Editor (`bedit` command).

Using Fields with Attributes

Fields allow you to place information in your drawing based on a graphical or nongraphical drawing object, the current value of a system variable, a property of a sheet set or project, the current date or the date when the current drawing was created, and much more. Fields can be used to construct the string value of a single-line or multiline text object, which can be a standalone object or in another object such as a table or dimension. In addition to text objects, attributes can use fields to define the value that they hold.

Use the following steps to define an attribute definition with a field:

1. Create an attribute definition as explained in the earlier section "Adding an Attribute Definition."
2. Instead of adding a static default value to the Default text box, click Insert Field (located to the right of the text box).
3. In the Field (Windows) or Insert Field (Mac OS) dialog box, select a Field Name from the leftmost column.
4. Select any options for the selected field.
5. Click OK (Windows) or Choose (Mac OS) to add the field definition to the Default text box. If you need to make a change to the field, double-click the shaded text in the text box. Click before or after the field definition to add some static text as needed.
6. Finish creating the attribute as you normally would.

When you insert a block reference with an attribute, you can specify a field value in the Edit Attributes dialog box by right-clicking in the text box to the right of the attribute prompt and clicking Insert Field. You can edit an existing field by double-clicking the shaded text in the text box, or you can convert it to static text by selecting and right-clicking the shaded text that represents the field and clicking Convert Field To Text.

BlockPlaceholder Field Type

In AutoCAD on Windows while the Block Editor is active, you have access to an additional field type: BlockPlaceholder. The BlockPlaceholder field type allows you to access the properties of the block reference when it is placed in a drawing. For example, you could list the name of the block as part of a description in an attribute field or even access the current value of one of the parameters used for a custom dynamic property.

Adding Dynamic Properties to a Block Definition

Dynamic properties when added to a block definition allow you to rotate, move, stretch, and perform other actions on the objects within a block reference. A block definition that contains dynamic properties is known as a *dynamic block*. Parameters and actions are used to implement dynamic properties. You can only add dynamic properties to a block definition using the Block Editor in AutoCAD on Windows. Although you can't create or modify dynamic properties in AutoCAD for Mac OS, you can insert blocks that have these properties already implemented.

When a block with dynamic properties is inserted into a drawing, additional grips are displayed for the block reference when it is selected. These additional grips allow you to interactively change the values of the custom actions rather than selecting values from a predefined

list. Figure 3.7 shows a block reference selected in the drawing and a linear distance being modified with grips.

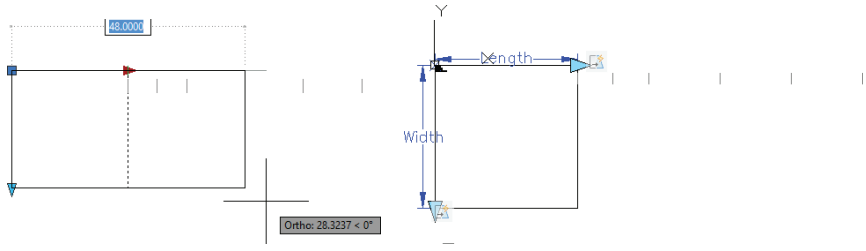


FIGURE 3.7 Modifying a block's dynamic properties with grips (left: in the drawing window; right: in the Block Editor)

Using Parameters to Modify Geometry in a Block Definition (Windows Only)

Parameters in a dynamic block define how geometry within a block reference can be modified. There are 10 different parameters that you can add to a block definition, and most of them must be paired with an action before they can modify the objects in a block reference. You create parameters by using the `bparameter` command, and the command can only be used when the Block Editor is active. While the `bparameter` command works from the command prompt, you can modify many of the properties of a parameter by using grips and the Properties palette (`properties` command).

The following lists the available parameters along with the actions they can be paired with:

Point The Point parameter defines a point in the block that can be used to move or stretch the geometry associated with that coordinate value. **Actions:** Move and Stretch.

Linear This parameter is used to modify geometry along a linear path. **Actions:** Array, Move, Scale, and Stretch.

Polar The Polar parameter is used to modify geometry along a polar path. **Actions:** Array, Move, Scale, Stretch, and Polar Stretch.

XY Use the XY parameter to modify geometry in both a horizontal and vertical direction. **Actions:** Array, Move, Scale, and Stretch.

Rotation This parameter rotates geometry around a given point. **Action:** Rotation.

Alignment Use this parameter to align a block perpendicular or tangent to the objects it is inserted near. **Action:** No action required.

Flip The Flip parameter mirrors the block reference and all objects in it. **Action:** Flip.

Visibility When you need to control the use of visibility states for the geometry within a block reference, add the Visibility parameter. **Action:** No action required.

Lookup The Lookup parameter creates a mapping table between a list of values and custom properties. **Action:** Lookup.

Base Point The Base Point parameter redefines the block definition's insertion point. **Action:** No action required.

Adding a Parameter to a Block Definition

The following example explains how to add a Linear parameter to a block definition:

1. Create or open an existing block definition, as I discussed in the "Creating or Editing a Block Definition" section earlier in this chapter.
2. On the Block Authoring Palettes window, click the Parameters tab, and then click Linear. If the Block Authoring Palettes window is not displayed, on the ribbon click Block Editor tab ➢ Manage panel ➢ Authoring Palettes.
3. At the `Specify start point or [Name/Label/Chain/Description/Base/Palette/Value set]:` prompt, specify a point in the drawing. Typically, this point is located using an object snap, such as an Endpoint, Intersection, or Center.
4. At the `Specify endpoint:` prompt, specify the endpoint for the parameter. Typically, this point is located using an object snap, such as an Endpoint, Intersection, or Center.
5. At the `Specify label location:` prompt, specify a location for the label for the parameter. You should notice a small yellow icon displayed near the parameter; this indicates that the parameter requires an action. The label is displayed in the Block Editor, Properties palette (Windows), and Properties Inspector (Mac OS) when you are modifying the block definition or a block reference. The label is also used when you want to reference the custom property with a field.
6. In the drawing window, select the parameter that you just placed. In the Properties palette, change the properties in the Property Labels, Geometry, Value Set, and Misc areas to control the behavior of the parameter. If the Properties palette is not displayed, select the parameter, right-click in the drawing window, and click Properties.
7. On the ribbon, click Block Editor tab ➢ Open/Save panel ➢ Save Block.
8. If the Block - Save Parameter Changes task dialog box is displayed, click Save The Changes.

Associating an Action with a Parameter (Windows Only)

Actions define which objects are controlled by a parameter and how those objects should be modified. You must add a parameter to a block definition before you can assign an action. There are eight different actions that you can pair with seven different parameters; some actions only work with one type of parameter whereas other actions can be used with several types of parameters. Refer to the earlier section "Adding Parameters to a Block Definition (Windows Only)" for information about which actions can be used with each parameter.

Actions are created using the `bactiontool` command; the command can only be used when the Block Editor is active. While the `bactiontool` command works from the command prompt, you can modify the properties of an action by using grips or the Properties palette (`properties` command).

Adding an Action to a Parameter in a Block Definition

The following steps explain how to add a Stretch action to a Linear parameter in a block definition:

1. Open an existing block definition that contains a Linear parameter. I explained how to add a Linear parameter to a block definition in the section "Adding Parameters to a Block Definition (Windows Only)" earlier in this chapter.
2. In the Block Authoring Palettes window, click the Actions tab and then click Scale. (If the Block Authoring Palettes window is not displayed, on the ribbon click Block Editor tab ➢ Manage panel ➢ Authoring Palettes.)
3. At the `Select parameter:` prompt, select the linear parameter.
4. At the `Specify parameter point to associate with action or enter [sTart point/Second point] <Second>:` prompt, select the grip point on the Linear parameter that should be associated with the Stretch action.
5. At the `Specify first corner of stretch frame or [CPolygon]:` prompt, specify the first corner of the crossing window for the Stretch action.
6. At the `Specify opposite corner:` prompt, specify the second corner of the crossing window for the Stretch action.
7. At the prompt

   ```
   Specify objects to stretch
   Select objects:
   ```

   specify the objects inside or that cross the Stretch frame you want to stretch or move. Stretch behavior is just like that of the `stretch` command. Press Enter to end object selection.
8. In the drawing window, select the action that you just added.
9. On the Properties palette, change the properties in the Stretch Frame, Overrides, and Misc areas to control the behavior of the action. (If the Properties palette is not displayed, select the parameter, right-click in the drawing window, and click Properties.)
10. On the ribbon, click Block Editor tab ➢ Open/Save panel ➢ Test Block.
11. In the drawing window, select the block and click the grip on the Linear parameter. Notice that you can stretch objects that cross the Stretch frame and move those that are completely inside the frame.
12. On the ribbon, click *<current>* tab ➢ Close panel ➢ Close Test Block Window to return to the Block Editor.

You can remove an action by right-clicking over its icon that is displayed on the action bar in the drawing and clicking Delete. The objects associated with an action can also be updated by using the options under the Action Selection Set submenu that is displayed when you right-click an action's icon on the action bar. An action's selection set can also be updated from the Misc area of the Properties palette. If an action's icon is not displayed in the drawing area, make

sure the `bactionbarmode` system variable is set to `1`. You might also need to use the `bactionbar` command and select the parameter that contains the action you want to modify.

NOTE The Parameter Sets tab of the Block Authoring Palettes window contains commonly used parameter and action groupings. These tools allow you to place a parameter and an action as a single operation instead of starting multiple commands.

Creating a Lookup Table (Windows Only)

Lookup tables allow you to map the values of one or more parameters to a more meaningful label. For example, you might have a listing of bolt-head sizes or lengths, or maybe a number of different sizes for doors and windows that you use in your drawings. Choosing an item from a lookup list sets the parameter values accordingly based on the value mappings instead of you having to specify multiple parameter values individually.

Creating a lookup table requires you to have one or more parameters already defined in your block definition, and then a Lookup parameter and action must be added to a block definition with the Block Editor. You add items to a lookup table using the Property Lookup Table dialog box (see Figure 3.8). After a block reference is inserted that contains a lookup table, you can choose an item from the Lookup grip, or use the drop-down list that is displayed in the Custom area of the Properties palette (Windows) or Properties Inspector (Mac OS) when a block reference is selected.

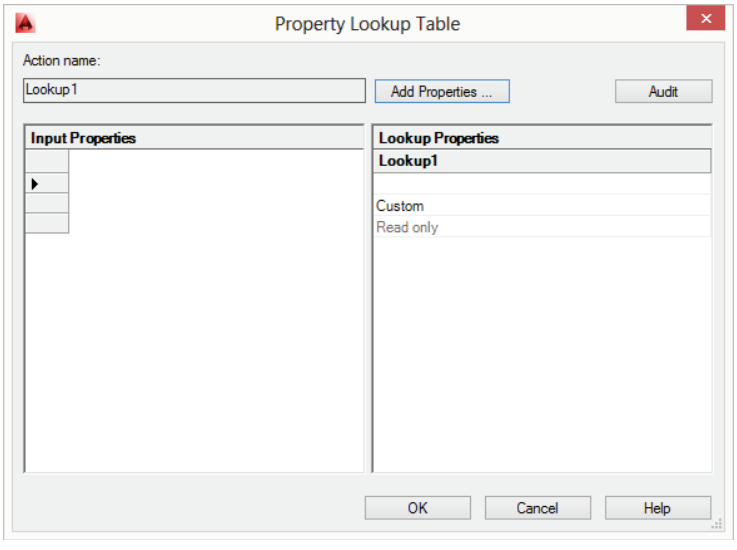


FIGURE 3.8 Defining a lookup table for a dynamic block

Adding a Lookup Table to a Block Definition

You can use these steps to create a lookup table and map parameter values to textual values for a block definition:

1. Open an existing block definition that contains one or more parameters. I explained how to add a Linear parameter to a block definition in the section "Adding Parameters to a Block Definition (Windows Only)" earlier in this chapter.
2. In the Block Authoring Palettes window, click the Parameter Sets tab, and scroll down to and click Lookup Set. (If the Block Authoring Palettes window is not displayed, on the ribbon click Block Editor tab ➢ Manage panel ➢ Authoring Palettes.)
3. At the `Specify parameter location or [Name/Label/Description/Palette]:` prompt, specify a point on or near the geometry of the block. Make sure not to place the parameter at the origin of the block or in the same location as another parameter.
4. Right-click over the new Lookup action icon and click Display Lookup Table to display the Property Lookup Table dialog box (`blookuptable` command).
5. In the Property Lookup Table dialog box, click Add Properties.
6. In the Add Parameter Properties dialog box, select the parameter you want to use with the lookup table and click OK. The Property section at the bottom of the dialog box filters out the parameters that are available to be added or that are already in use with the lookup table.
7. In the Input Properties area, in the column of the parameter you just added, enter the parameter value that the lookup item should be mapped to. For example, if the parameter represents the width of a door or window, you would enter the width for that item, such as 32.000 or 36.000.
8. In the the Lookup Properties area, in the column of the lookup parameter you are working with, enter a textual value for each parameter value you entered. For example, if 32.000 and 36.000 were entered for door widths you might enter the textual strings **`32"W Door`** and **`36"W Door`**.
9. Click OK to return to the Block Editor.
10. On the ribbon, click Block Editor tab ➢ Open/Save panel ➢ Test Block.
11. In the drawing area, select the block and click the grip for the Lookup parameter. Notice the values that are listed. Select one or more of the options from the list to make sure your parameters are being updated correctly.
12. On the ribbon, click *<current>* tab ➢ Close panel ➢ Close Test Block Window to return to the Block Editor.

Defining Visibility States (Windows Only)

The geometry of a block definition, as you have previously read, can be resized, rotated, stretched, flipped, arrayed, and moved using parameters and actions; visibility states do not modify the objects in a block reference, but rather control which objects are displayed based on which state is current. For example, you could have a single block definition with multiple visibility states that allow you to display an open or closed valve based on which option you choose. You could also use visibility states to toggle the view of a bolt; plan and side views could be different visibility states of the same block.

Creating a visibility state requires you to add a Visibility parameter to a block definition with the Block Editor, and then you can modify which visibility states are available using the Visibility States dialog box (see Figure 3.9). A visibility state can be chosen from the Visibility grip of a block reference in a drawing, or you can select a value from the drop-down list that is displayed in the Custom area of the Properties palette when a block reference is selected.

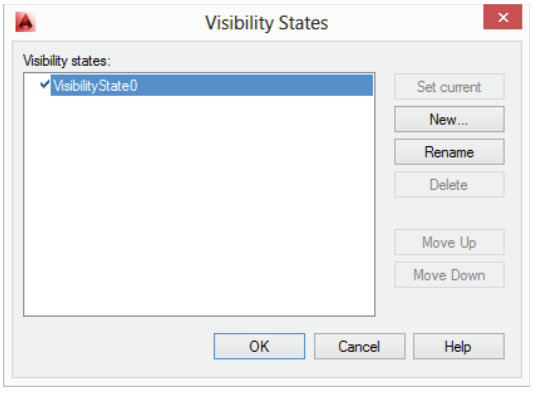


FIGURE 3.9 Defining a lookup table for a dynamic block

The following steps explain how to add a Visibility parameter to a block definition and control the objects of a visibility state:

1. Create or open an existing block definition as I discussed in the "Creating or Editing a Block Definition" section earlier in this chapter.
2. On the Block Authoring Palettes window, click the Parameters tab and then click Visibility. (If the Block Authoring Palettes window is not displayed, on the ribbon click Block Editor tab ➢ Manage panel ➢ Authoring Palettes.)
3. At the `Specify parameter location or [Name/Label/Description/Palette]:` prompt, specify a point on or near the geometry of the block. Do not place the parameter at the origin of the block or in the same location as another parameter.
4. Double-click the new Visibility parameter to display the Visibility States dialog box (`bvstate` command).
5. In the Visibility States dialog box, click Rename to change the name of the default visibility state, VisibilityState0. Enter a new name for the visibility state.
6. Click New to display the New Visibility State dialog box.
7. In the New Visibility State dialog box, enter a name for the new visibility state in the Visibility State Name text box. Choose an option in the Visibility Options For New States section to control the default visibility of the objects in the block definition for the new visibility state. Click OK.
8. In the Visibility States dialog box, select the visibility state that you want to work with in the Block Editor. You can also click Move Up or Move Down after selecting a visibility

state to control its order in the drop-down list on the Visibility panel of the Block Editor tab on the ribbon or in the Visibility parameter's list when a block reference is inserted.

9. On the ribbon, click Block Editor tab ➢ Visibility panel, and then click Make Visible (`bvshow` command) or Make Invisible (`bvhide` command) to control the visibility of the geometry in the block definition for the current visibility state. You can also click Visibility Mode (`bvmode` system variable) to show all hidden objects in the current visibility state so they can be shown using the Make Visible button.
10. On the ribbon, click Block Editor tab ➢ Open/Save panel ➢ Test Block.
11. In the drawing window, select the block and click the grip for the Visibility parameter. Notice the values that are listed. Select one or more of the options from the list to make sure the objects in each visibility state are being displayed correctly.
12. On the ribbon, click *<current>* tab ➢ Close panel ➢ Close Test Block Window to return to the Block Editor.

Inserting and Modifying Dynamic Blocks

Blocks with dynamic properties, or *dynamic blocks*, can be inserted using the same methods that are used to insert any other block definition into a drawing. All of the methods that I mentioned earlier, in the section "Inserting or Replacing a Block Reference," can be used to insert a dynamic block.

I have a few favorite tricks that you might find helpful when inserting dynamic blocks:

- In AutoCAD on Windows, when you use the `insert` command, you can control the custom properties of a dynamic block while it is being inserted with the Properties palette.
- While the Block Editor is active and a dynamic block is being inserted, you can press Ctrl to cycle through grips of the parameters in the block and use those locations as possible insertion points of the block reference. The order in which the grips are cycled through is controlled by the Insertion Cycle Order dialog box (`bcycleorder` command).
- In AutoCAD on Windows, if you add a dynamic block as a tool on a tool palette for the Tool Palettes window, you can control the default values used for the block's custom properties by right-clicking over the tool and clicking Properties. When the Tool Properties dialog box opens, you can set the desired values.

After a dynamic block has been inserted into a drawing, you commonly change its custom properties interactively using the grips that are displayed in the drawing area when the block reference is selected. If grips for the dynamic properties are not displayed, make sure the `grips` system variable is set to a value of `1` or `2`. You can also use the Properties palette (Windows) or Properties Inspector (Mac OS) to change the custom properties of the dynamic block and the other general properties of the block reference.

If the dynamic block contains attribute definitions, they behave just like they do in standard blocks with attributes. The only differences might be that the placement of an attribute within a block reference might change based on the way the actions in the dynamic block were defined. You can use the Lock Position mode for an attribute definition to force that attribute to be

ignored by any actions that might have the attribute definition in their selection set. The Default property value of attribute definitions might also be assigned a `BlockPlaceholder` field type; this allows the attribute to inherit the current value of any custom property within the block reference.

Extracting Properties and Values from a Block Reference

One of the benefits of using attributes with your blocks is that you can extract the information and then use it in your drawing or an external program. If all of your blocks contain attributes, you could generate a bill of materials on demand or even help prepare an estimate for a project. The attribute values and the properties of the blocks containing attributes can be extracted from a drawing using the Attribute Extraction dialog box (`attext` command) or Data Extraction Wizard (`attext` command) (in AutoCAD on Windows only) or by using the `-attext` command at the command prompt in AutoCAD both on Windows and Mac OS.

The following steps explain how to extract the attributes from a drawing using the Data Extraction Wizard in AutoCAD on Windows:

1. At the command prompt, enter **`dataextraction`** and press Enter.
2. If the Data Extraction - Unsaved Drawing message is displayed, click Save to continue.
3. On the Begin page in the Data Extraction Wizard, click Create A New Data Extraction to create a new data extraction (DXE) file. Click Next. When creating a new file, you can choose an existing DXE file to use as a template if you have one available. You can also use an existing DXE file instead of starting from scratch.
4. In the Save Data Extraction As dialog box, enter a name for the new DXE file in the File Name text box. Click Save.
5. In the Data Source section on the Define Data Source page, click Drawings/Sheet Set, and then click one of the following:
   - Drawings/Sheet Set: This option allows you to extract attributes from the current drawing, one or more additional drawings, or all drawing files in a folder. Click Add Folder to add all drawings from a folder or click Add Drawings to select specific additional drawings.
   - Select Objects In The Current Drawing: This selection allows you to extract attributes from specific objects in the current drawing. Click the Select Objects In The Current Drawing button to select the objects in the drawing and press Enter to end object selection.
6. Click Settings to display the Data Extraction - Additional Settings dialog box. Specify the extraction options you want to use and click OK. Click Next.
7. In the Objects section on the Select Objects page, clear the check boxes for the objects you do not want to include in the extraction process. You can also use the options in the Display Options section to filter the Objects list to only blocks or non-block objects. Clear Display All Object Types, and check both Display Blocks Only and Display Blocks With Attributes to extract block attribute information only. Click Next.

8. In the Properties section on the Select Properties page, clear the check boxes for the properties you do not want to include in the extraction process. You can remove an entire category of properties by clearing check boxes in the Category Filter section. Click Next.
9. On the Refine Data page, choose how you want the information to be extracted. Right-click over the data grid for additional data-formatting options, the ability to sort and filter data in columns, and the ability to rename or hide columns. Click Full Preview to see a preview of the extracted attributes and the options specified. Close the preview window. Click Next.
10. On the Choose Output page, click Insert Data Extraction Table Into Drawing to create a table object that contains the extracted attributes or click Output Data To External File to export the extracted attributes to a file that can be imported into another application. If you click Output Data To External File, you can click the ellipsis […] button to specify a name and format for the external file. Click Next.
11. If you checked Insert Data Extraction Table Into Drawing, the Table Style page is displayed. Specify the table style you want to use and how the table should be structured. Click Next.
12. On the Finish page, click Finish. If you checked Insert Data Extraction Table Into Drawing, specify an insertion point for the table. If you checked Output Data To External File, the external file is created.

NOTE After you place a table using the Data Extraction Wizard, a Data Link icon that looks like two chain links appears in the application's status-bar tray. Right-click the icon or the table in the drawing, and click Update All Data Links to ensure the information displayed in the table is up to date.

If you are using AutoCAD for Mac, you can extract attribute values only with the `-attext` command; the command is also available in AutoCAD on Windows. The `-attext` command allows you to extract attribute values to an external file, which can then be imported into any application that can read data from a comma- or space-delimited file. It is also possible to use an AutoLISP program to generate a table object and populate it with the exported attribute information. You first need to create an attribute-extraction template, which is a TXT file that defines the data structure to be extracted.

Each line in an attribute-extraction template file must follow one of these syntaxes:

```
<Block Property Name or Attribute Tag> Nwwwddd
<Block Property Name or Attribute Tag> Cwww000
```

The following explains the syntax used in an attribute-extraction template file:

`<Block Property Name or Attribute Tag>` This variable specifies the name of the block property or attribute tag you want to extract information from in your drawing. If a block does not contain the attribute tag specified, no error is generated and AutoCAD continues to look for blocks that have attributes that match those defined in the template.

`N` Use `N` to specify that the extracted value should be formatted as a numeric value.

`C` Use `C` to specify that the extracted value should be formatted as a character string value.

`www` Replace the *`www`* variable with a fixed length (number of characters) for the extracted value. Add 000 (three zeros) after the fixed length value. 000 doesn't impact how attribute values are extracted or formatted, but does affect how the attribute-extraction template file is parsed. Not including 000 could result in unexpected problems.

ddd Replace the *ddd* variable with the number of decimal places to use for extracted numeric values. The attribute can't contain non-numeric values, such as quotation marks. If an attribute value is not numeric, the message `** Bad numeric value for field <name>, record n` is displayed in the command-line window.

Table 3.1 lists the block properties that can be used in an attribute-extraction template and describes each.

TABLE 3.1: Block properties available for use in an attribute-extraction template

BLOCK PROPERTY	DESCRIPTION
`BL:NAME`	Name of the block reference
`BL:LEVEL`	Nesting level of the block reference
`BL:X`, `BL:Y`, and `BL:Z`	X, Y, and Z coordinate values of the block reference
`BL:NUMBER`	Counter for the block reference
`BL:HANDLE`	Handle for the block reference
`BL:LAYER`	Layer the block reference is placed on
`BL:ORIENT`	Rotation of the block reference
`BL:XSCALE`, `BL:YSCALE`, and `BL:ZSCALE`	X, Y, and Z scale-factor values of the block reference
`BL:XEXTRUDE`, `BL:YEXTRUDE`, and `BL:ZEXTRUDE`	X, Y, and Z extrusion-direction values of the block reference

The following steps explain how to define an attribute-extraction template:

1. In the Mac OS Finder, click Go ➢ Applications. In the Finder window, double-click TextEdit.
2. In TextEdit, click TextEdit ➢ Preferences. In the Preferences dialog box, on the New Document tab click Plain Text and then close the dialog box.
3. Click File ➢ New to create a plain ASCII text file.
4. Click File ➢ Save and enter a name for the file in the Save As text box. Specify a location for the file and click Save. Keep the file extension as `.txt`.
5. In the TextEdit window, enter the data-structure elements for the attribute-extraction template. For example, the following extracts the name of each block reference and the attribute tags `WIDTH` and `DESCRIPTION`:

```
BL:NAME     C015000
WIDTH       N010000
DESCRIPTION C050000
```

6. Click File ➢ Save to save the changes to the attribute-extraction template.

NOTE If you are using AutoCAD on Windows, you can use Notepad instead of TextEdit to create an attribute-extraction template.

The following steps explain how to extract attribute values using an attribute extraction template and the `-attext` command:

1. In AutoCAD, open the drawing that contains the blocks and attribute values you want to extract.
2. At the command prompt, enter **`-attext`** and press Enter.
3. At the `Enter extraction type or enable object selection [Cdf/Sdf/Dxf/Objects] <C>:` prompt, enter **`c`** and press Enter.
4. In the Select Template File dialog box, browse to and select the attribute-extraction template you want to use. Click Open.
5. In the Create Extract File dialog box, enter a name and specify a location for the file. Click Save.
6. In Finder, browse to and open the file created in the previous step. The following shows what some of the extracted data might look like for a drawing that contains a block named `RDESK`:

```
'RDESK', 30, 'Rectangular Desk with Beveled Edging'
'RDESK', 24, 'Rectangular Desk with Beveled Edging'
'RDESK', 24, 'Rectangular Desk with Beveled Edging'
'RDESK', 30, 'Rectangular Desk with Beveled Edging'
```

Chapter 4

Manipulating the Drawing Environment

The chapters until now have focused primarily on drawing-file customization: establishing and enforcing drawing standards, creating drawing templates, and working with named objects and blocks. While drawing-file customization should be used to form the cornerstone of your overall customization plans for the Autodesk® AutoCAD® program, there are limitations to the productivity gains that can be made with drawing-file customization only.

After you have established the drawing standards and template files that you will use to create new drawings, the next step in becoming more productive is to customize the AutoCAD drawing environment. The drawing environment controls the settings that are used to specify the behavior of commands, where AutoCAD looks for custom files, how the user interface looks, the behavior of the command prompt, and much more.

Getting Started with Drawing-Environment Customization

Although it might be new to you, customizing AutoCAD is something with which you should feel free to experiment. Before you start, I have a few words of advice:

- Back up the AutoCAD customizable files before making changes to them. In Windows, use the product's Export AutoCAD *<release>* Settings option on the Start menu or Start screen to create a backup of the custom settings and files for AutoCAD. You can then restore the exported settings using the Import AutoCAD *<release>* Settings option on the Start menu or Start screen. On Mac OS, you will need to copy the files under the `Local` and `Roaming` folders located under `/Users/<user_name>/Library/Application Support/Autodesk` on your local hard drive.
- Start small and test often. Too often I have seen people take on a customization project that is more than they are ready for and get frustrated, which has led them to just give up. You have to learn to walk before you can run. Also, make sure you do not customize lots of different files and features in AutoCAD without testing your changes; making too many changes at once could lead to some unnecessary troubleshooting that could have been avoided if additional testing was performed along the way.

- If you get stuck, take a deep breath and reset AutoCAD to its installed state. You will want to try and restore any files that you previously backed up before resetting AutoCAD, though. Even before resetting AutoCAD, make a copy of the current customized files. On Windows, use the product's Reset Settings To Default option on the Start menu or Start screen to reset the program's custom settings and files. On Mac OS, start AutoCAD and display the Application Preferences dialog box (`options` command). In the Application Preferences dialog box, on the Application tab click Reset Application Options and then click Reset AutoCAD.

Now that you are ready to get started, here are the ways the drawing environment can be customized that we'll discuss in this chapter:

- Command-line switches can be used in combination with desktop icons or batch and Bash shell files to control the way AutoCAD starts up.
- Preferences and settings that affect the way the application looks, where custom files can be located, and how some commands behave can be set and managed with the `options` command.
- System and environment variables can be queried and set from the command prompt to control the behavior of AutoCAD and many commands.
- Command aliases and external commands can be defined to allow quicker access to commands and system variables from the command prompt.
- Command-line input can be configured to control the way typed input is handled and the choices that are displayed in the command suggestion list.

Customizing the AutoCAD Startup Process

Before you start AutoCAD, you have an opportunity to specify which drawing to open and additional parameters known as *command-line switches* that can be used to control a number of different features at startup. Command-line switches must be placed after the `acad.exe` file in the Target field of a shortcut or batch file on Windows, or after `AutoCAD` in a Bash shell file on Mac OS.

The following shows an example of calling `acad.exe` on Windows using the `/t` command-line switch to create a new drawing based on a drawing template file named `acad3D.dwt`:

```
"C:\Program Files\Autodesk\AutoCAD 2014\acad.exe" /t "acad3D.dwt"
```

The following shows an example of calling `AutoCAD` on Mac OS using the `-t` command-line switch to create a new drawing based on the `acad3D.dwt` drawing template file:

```
"/Applications/Autodesk/AutoCAD 2014/AutoCAD 2014.app/Contents/MacOS/AutoCAD" ↵
-t "acad3D.dwt"
```

AutoCAD on Windows supports 15 command-line switches, whereas AutoCAD on Mac OS supports only 5. You can use multiple command-line switches when starting AutoCAD: just add a space after a switch and the previous parameter. Table 4.1 lists the most commonly used command-line switches for Windows, as well as 3 of the 5 switches that are supported on Mac OS.

TABLE 4.1: AutoCAD command-line switches

Switch Name (Windows)	Switch Name (Mac OS)	Description
/b	-b	Starts the script file after the specified or default drawing is opened at startup **Usage:** /b "script_name.scr" or -b "script_name.scr"
/nohardware		Disables hardware acceleration **Usage:** /nohardware
/nologo	-nologo	Hides the application's splash screen **Usage:** /nologo or -nologo
/p		Sets a user profile as current or imports a previously exported user profile **Usage:** /p "profile_name|profile.arg"
/t	-t	Creates a new drawing file based on the specified drawing template file **Usage:** /t "template_name.dwt" or -t "template_name.dwt"
/set		Loads a Sheet Set (DST) file **Usage:** /set "sheet_set_name.dst"
/w		Sets a workspace as current **Usage:** /w "workspace_name"

NOTE You can search AutoCAD help on the keywords "command line switch reference" to learn about all of the available switches.

On Windows, you can create a new desktop shortcut that starts AutoCAD and specifies the use of more than one switch. The example that follows uses the /t, /p, and /w command-line switches. Begin by copying the book's sample files from www.sybex.com/go/autocadcustomization to a folder named MyCustomFiles.

NOTE This example assumes that you copied the samples files for the book to a folder named MyCustomFiles in your Documents (or My Documents) folder. If you placed the sample files in a different folder, you will need to make the appropriate changes to the value entered in step 7.

1. Minimize all open applications until you see the Windows Desktop. (You can press the Windows+D key combination to minimize all open windows and then repeat the key combination to restore all the applications to their previous display state.)

2. On the Windows Desktop, right-click and then click New ➤ Shortcut from the menu that opens.

WARNING Based on the way your workstation or Windows login has been configured, you might be restricted from adding or modifying shortcuts on the Windows Desktop. You might need to create the new shortcut in a different location, such as the `MyCustomFiles` folder. Contact your system administrator for information on your company's specific policies.

3. When the Create Shortcut dialog box opens, click Browse.
4. In the Browse For Files Or Folders dialog box, expand Computer and browse to `C:\Program Files\Autodesk\AutoCAD` *<release>*. Then select `acad.exe` and click OK. You are returned to the Create Shortcut dialog box (see Figure 4.1), and the location of the `acad.exe` file is displayed.
5. Click in the Type The Location Of The Item text box, and press the End key.

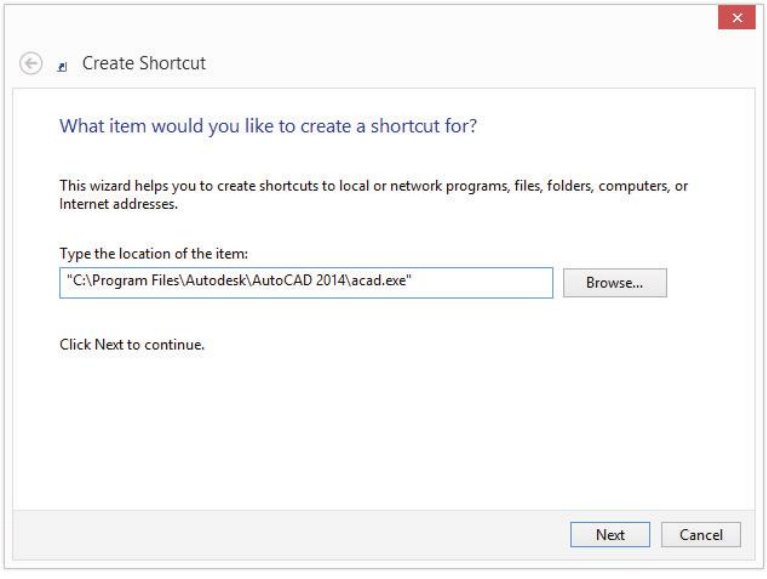


FIGURE 4.1 Setting the properties for the new shortcut

6. With focus still in the Type The Location Of The Item text box, press the spacebar once and then enter the following:

```
/t "%userprofile%/Documents/MyCustomFiles/C-Size.dwt" /p ↵
"<<Unnamed Profile>>" /w "AutoCAD Classic"
```

NOTE Filenames and folder paths that contain spaces must begin and end with a double quotation mark (").

7. Click Next.
8. In the Type A Name For This Shortcut text box, clear the current value and then type **Classic AutoCAD**.
9. Click Finish to create the shortcut. If prompted to replace the shortcut, click No and enter a new name. Then click Finish.
10. Double-click the original AutoCAD shortcut on your desktop. At the command prompt, type **`wscurrent`** and press Enter. Write down the value within the angle brackets and press Enter again. At the command prompt, type **`cprofile`** and press Enter. Write down the value after the equals sign. You now have recoded the current workspace and profiles so you can restore them later.
11. Double-click the Classic AutoCAD shortcut. The default drawing created will be based on the `C-Size.dwt` file, and instead of the ribbon, you should see the classic menu bar and toolbars.
12. Double-click the original AutoCAD shortcut on your desktop. The default drawing will no longer be based on the `C-Size.dwt` file, but the user interface might not be correct until you do the following:
    - From the Workspaces drop-down list on the Quick Access toolbar (see Figure 4.2), select the value of the `wscurrent` system variable that you wrote down in step 10.
    - Right-click in the drawing window and click Options. In the Options dialog box, click the Profiles tab and then select the profile you wrote down in step 10. Click Set Current and then click OK to set the profile as current and exit the Options dialog box.

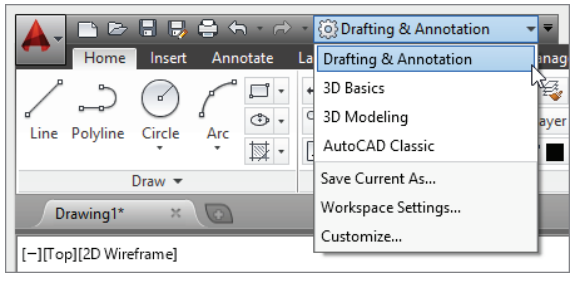


FIGURE 4.2 Switching workspaces

After you create a shortcut, you can edit the command-line switches that it uses. To edit a shortcut, right-click the shortcut and click Properties. When the Properties dialog box opens, click the Shortcut tab, and edit the text in the Target text box.

On a system running Mac OS, you can use the following steps to create a Bash shell file named `acad-startup.sh` that can be used to start AutoCAD and specify the use of the `-nologo` and `-t` command-line switches. Begin by copying the book's sample files from `www.sybex.com/go/autocadcustomization` to a folder named `MyCustomFiles`.

NOTE This example assumes that you copied the sample files for the book to a folder named `MyCustomFiles` in your `Documents` folder. If the sample files were placed in a different location, you will need to change the values entered in steps 5 and 8.

1. In the Mac OS Finder, click Go ➤ Applications. In the Finder window, double-click TextEdit.
2. In TextEdit, click TextEdit ➤ Preferences. In the Preferences dialog box, on the New Document tab click Plain Text and then close the dialog box.
3. Click File ➤ New to create a plain ASCII text file.
4. Click File ➤ Save and type **`acad-startup.sh`** in the Save As text box. From the sidebar on the left, click Documents ➤ MyCustomFiles and then click Save.
5. In the TextEdit window, enter the following text on one line:
   `"/Applications/Autodesk/AutoCAD 2014/AutoCAD 2014.app/Contents/MacOS/ ↵ AutoCAD" -nologo -t ~/Documents/MyCustomFiles/C-Size.dwt`

NOTE Filenames and folder paths that contain spaces must begin and end with a double quotation mark (“).

6. Click File ➤ Save to save the changes to the shell file.
7. In the Mac OS Finder, click Go ➤ Utilities. In the Finder window, double-click Terminal.
8. In the Terminal window, type **`chmod u+x ~/Documents/MyCustomFiles/acad-startup.sh`** and press Enter. Without changing the permissions of the file, the file can’t be executed in the Terminal Window.
9. In Finder, browse to the `acad-startup.sh` file. Press and hold the Control key, and then click over the `acad-startup.sh` file. You could also two-finger-click or right-click over the file, based on how your mouse or trackpad is configured.
10. From the shortcut menu, click Open With ➤ Other.
11. In the Choose Application dialog box, scroll down to and double-click Utilities. Click the Enable drop-down list and select All Applications. Click the Always Open With check box and select Terminal. Click open.
12. Double-click the Bash shell file to launch AutoCAD and have it create the default drawing based on the `C-Size.dwt` file.

WARNING The Terminal Window must remain open while AutoCAD is running because the AutoCAD process is attached to the Terminal Window. Closing the window will cause AutoCAD to crash.

Specifying Application Preferences

AutoCAD has evolved over the 30 or so years that it has been around, but the way users work with the software hasn’t always followed suit. The AutoCAD on Windows settings have been

organized into 12 categories to make it easy to locate a specific setting. (There are even more settings that are specific to AutoCAD-based vertical products.) AutoCAD on Mac groups its settings into 6 different categories. Many settings are available on both Windows and Mac OS, but there are many differences as well, because different features are supported on each platform.

You configure the application preferences using the `options` command, which displays the Options dialog box on Windows and the Application Preferences dialog box on Mac OS. Figure 4.3 shows both of the dialog boxes. In the Windows Options dialog box, the tabs along the top allow you to switch between settings categories. The vertical tabs on the left of the Mac OS Application Preferences dialog box perform the same function.

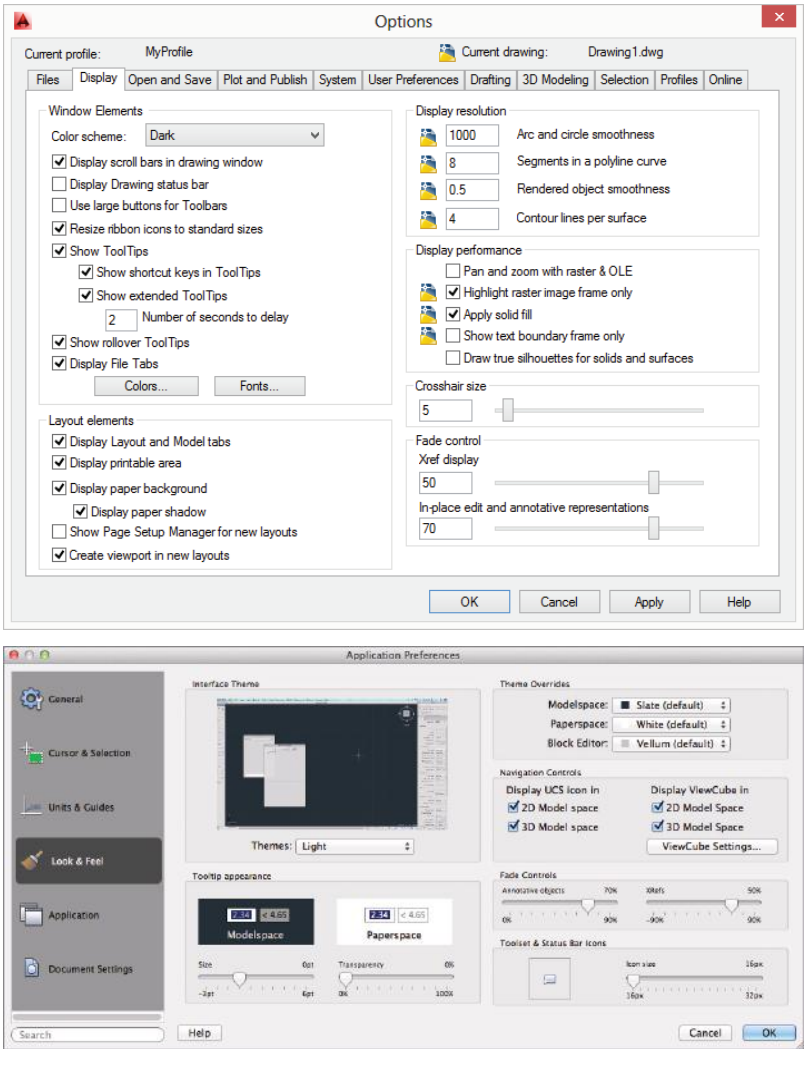


FIGURE 4.3
Setting application preferences

You can display the Options or Application Preferences dialog box using one of the following methods:

- Click the Application menu button ➢ Options (Windows).
- On the Mac OS menu bar, click AutoCAD <*release*> ➢ Preferences (Mac OS).
- Secondary-click or right-click in the drawing window or over the command-line window, and then click Options (Windows) or Preferences (Mac OS).
- At the command prompt, type **`options`** and press Enter (Windows and Mac OS).

Both of these dialog boxes offer an extensive number of settings that can't all be covered in this book. Some of them are user-driven choices that do not impact the use of menu elements, creation and running of scripts, or creation and deployment of AutoLISP programs or other custom programs. Throughout this book when appropriate, I will mention the settings that you should be aware of because they could alter the way you deploy your customization and explain how your customization might be affected. You can click the Help button in the Options or Application Preferences dialog box to learn more about the settings on each tab.

One of the main differences between AutoCAD on Windows and AutoCAD on a Mac OS is in how you can manage application preferences. AutoCAD on Windows supports multiple user profiles, whereas AutoCAD on Mac OS supports only a single user profile. User profiles allow you to configure AutoCAD several different ways, and then switch back and forth between the settings that you might need to use based on the type (2D/3D) or discipline (civil, architectural, etc.) of the drawing you are working on.

The following example explains how to create a user profile named `MyProfile` in AutoCAD on Windows:

1. Click the Application menu button ➢ Options.
2. In the Options dialog box, click the Profiles tab and then click Add To List.
3. When the Add Profile dialog box opens, type **MyProfile** in the Profile Name text box.
4. Optionally, enter a description in the Description text box.
5. Click Apply & Close.
6. Select the new profile from the Available Profiles list and click Set Current.
7. Switch to other tabs in the Options dialog box and make changes to the settings of the new profile.
8. Click OK to save the changes made to the profile.

A profile must be set as current before AutoCAD can use the settings it was defined with. The next steps explain how to set as current the `MyProfile` profile that you just created:

1. Click the Application menu button ➢ Options.
2. In the Options dialog box, click the Profiles tab.

3. On the Profiles tab, in the Available Profiles list, select MyProfile and click Set Current.
4. Click OK to exit the Options dialog box.

TIP You can also use the `/p` command-line switch, as explained in the section "Customizing the AutoCAD Startup Process" earlier in this chapter, to set a user profile as current when AutoCAD starts up.

User profiles can be shared with other users in your department or company. Use the following steps to export and then import the user profile `MyProfile` that you created earlier in this section:

1. Click the Application menu button ➢ Options.
2. In the Options dialog box, click the Profiles tab.
3. On the Profiles tab, in the Available Profiles list, select MyProfile and click Export.
4. In the Export Profile dialog box, enter **MyProfile** in the File Name text box and click Save. Specify a location other than `Documents` (or `My Documents`) if you want to, such as the `MyCustomFiles` folder that you created for this book.
5. Click Import to import the previously exported profile.
6. In the Import Profile dialog box, browse to and select the `myprofile.arg` file that you created in Step 4. Click Open.
7. In the Import Profile dialog box, keep the name in the Profile Name text box or enter a new one. For this example, type the name **`MyProfile-Imported`**.
8. Optionally, enter or change the description that is displayed. If you do not want to use the support paths and other paths defined in the exported user profile, clear the Include Path Information check box. Typically, you always want to use the paths that were exported with the user profile.
9. Click Apply & Close. Set the user profile as current if you want to use it.
10. Click OK to exit the Options dialog box.

TIP You can use the `/p` command-line switch to import and set a user profile as current when AutoCAD is started from a shortcut or batch file. I mentioned command-line switches in the section "Customizing the AutoCAD Startup Process" earlier in this chapter.

Customizing the Elements in the Drawing Window

The drawing window used to create and edit the objects in an open drawing file can be customized to show different elements. Some of the elements in a drawing window (see Figure 4.4) are designed for when you work on 3D models and others for when you are working on 2D drawings.

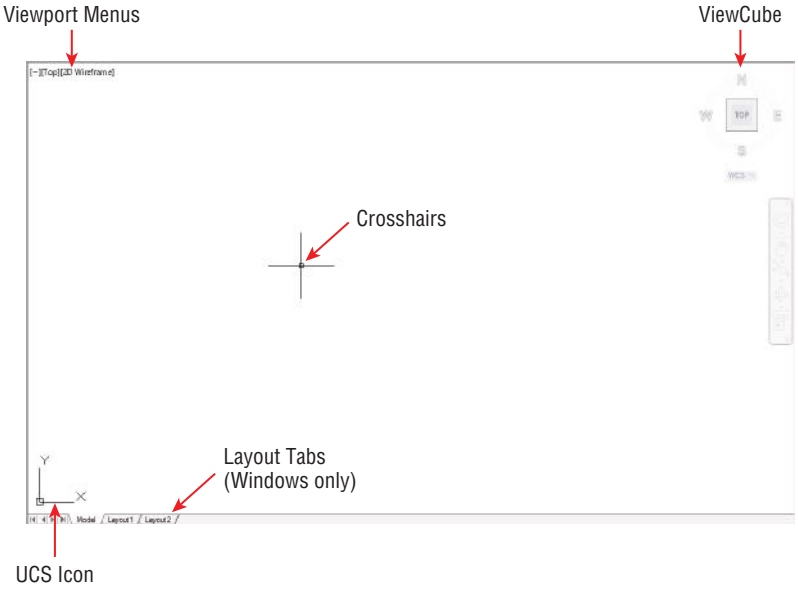


FIGURE 4.4 Elements of the drawing window

The following elements can be customized using the `options` command to display the Options dialog box (Windows) or Application Preferences dialog box (Mac OS):

Crosshairs The crosshairs provide you with a visual guide and an indicator of the location in the drawing window for the next point you pick. You can control the size of the crosshairs on the Display tab in the Options dialog box (Windows) or the Cursor & Selection tab in the Application Preferences dialog box (Mac OS). As an alternative to the dialog box, you can use the `cursorsize` system variable to change the size of the crosshairs. On Windows, you can also enable the display of the Z-axis and axes labels for the crosshairs on the 3D Modeling tab of the Options dialog box.

Viewport Menus Viewport menus allow you to access named and preset viewport configurations, views, and visual styles. You can also control the display of the ViewCube®, SteeringWheels®, and Navigation Bar. You can toggle the display of the Viewport menus on the 3D Modeling tab in the Options dialog box (Windows) or the General tab in the Application Preferences dialog box (Mac OS). The `vpcontrol` system variable can also be used to toggle the display of the Viewport menus.

UCS Icon The UCS icon shows the directions of the X-, Y-, and Z-axes in the current drawing. You can customize how the UCS icon looks using the Properties option of the `ucsicon` command. In addition to the appearance of the UCS icon, you can control if and where the UCS icon is displayed with the `ucsicon` command. The display of the UCS icon in a viewport with a 2D or 3D visual style can be controlled on the 3D Modeling tab in the Options dialog box (Windows) or the Look & Feel tab in the Application Preferences dialog box (Mac OS). As an alternative to using the dialog boxes, you can use the system variables that begin with `ucs` to control the behavior of the UCS icon as well as when it is displayed.

Layout Tabs (Windows Only) Layout tabs allow you to switch between Model and any named layouts that are defined in the drawing. When the tabs are suppressed, a new button

is added to the status bar. You can toggle the display of the Layout tabs on the Display tab in the Options dialog box.

ViewCube The ViewCube allows you to rotate the current view of the drawing by clicking one of the predefined areas or dragging the cube. The ViewCube is designed primarily for those who work on 3D models, but it does have some functionality that those working on 2D drawings can take advantage of. You can control the behavior and size of the ViewCube with the Settings option of the `navvcube` command, which displays the ViewCube Settings dialog box. The display of the ViewCube in a viewport with a 2D or 3D visual style can be controlled on the 3D Modeling tab in the Options dialog box (Windows) or the Look & Feel tab in the Application Preferences dialog box (Mac OS). As an alternative to using the dialog boxes, you can use `displayviewcubein2d` and other system variables that start with `navv` to display and control most of the features of the ViewCube.

Element Colors The colors for some of the elements in the drawing window are inherited from the operating system, but most of the colors used by elements in the drawing window are controlled by AutoCAD. The colors used by the drawing window can be controlled on the Display tab in the Options dialog box (Windows) or the Look & Feel tab in the Application Preferences dialog box (Mac OS). There is no alternative approach using system variables to manipulate the colors used in the drawing window, but you can use the ActiveX API with AutoLISP or VBA if you are using AutoCAD on Windows.

The changes made to these drawing-window elements are persisted between sessions as part of the AutoCAD user profile. AutoCAD for Mac does not support creating additional user profiles like AutoCAD on Windows does. For information on creating user profiles, see the "Specifying Application Preferences" section earlier in this chapter.

Configuring Command and Dynamic Input

An area of AutoCAD customization that is often overlooked is one that every user interacts with hundreds of times in a day—the command prompt. The command prompt is displayed in the command-line window and dynamic input features. When a command is not active, as you enter values at the command line or in a dynamic tooltip, you will see a suggested list of commands and system variables; the suggestions are based on the letters or numbers that you enter. You can customize the way you enter coordinate, angular, and distance values while a command is active and dynamic input is enabled.

Command Input Search Options

Starting with AutoCAD 2013, in Windows the Input Search Options dialog box (`inputsearchoptions` command) allows you to change the input settings for the command prompt (see Figure 4.5). In AutoCAD for Mac, use the `-inputsearchoptions` command; the Input Search options dialog box is not available. The `-inputsearchoptions` command is also available on Windows.

The command prompt features that you can control are as follows:

AutoComplete Controls whether AutoCAD attempts to match the letters or numbers users enter at the command prompt against all of the registered commands and system variables. Because of the way matching is performed, you see a larger number of entries in the suggestion list than when the feature is disabled. If you are using AutoCAD on Windows, you can

enable mid-string searching and control whether the suggestion list is sorted by usage or alphabetically.

AutoCorrect (AutoCAD 2014 on Windows Only) Allows AutoCAD to monitor the way you input commands and system variables. Over time, if you enter the same incorrect name more than three times and choose the correct one from the suggestion list, AutoCAD creates a command alias for you based on the incorrect name and the command or system variable you meant to enter. For more information on how the AutoCorrect feature works, see the section "Adding Synonym and AutoCorrect Entries (Windows Only)" later in this chapter.

Search for System Variables System variables are added to the suggestion list with commands. If you do not work with system variables all the time, disabling this feature can reduce the number of entries in the suggestion list for you. If you are using AutoCAD on Windows, you can separate commands and system variables in different areas of the suggestion list.

Suggestion-List Delay You can specify the number of milliseconds between keystrokes that AutoCAD delays before displaying the suggestion list. If you are a fast typist or the command prompt doesn't seem to recognize the keystrokes you type, you might want to set this number higher than its default of 300. You can also change the delay value by using the `inputsearchdelay` system variable.

Search Content (AutoCAD 2014 on Windows Only) Several of the named objects that can be created in a drawing can be set as current or used by entering the name of the object at the command prompt. If you do not set a named object as current or insert a block by typing its name at the command line, you might want to consider turning it off.

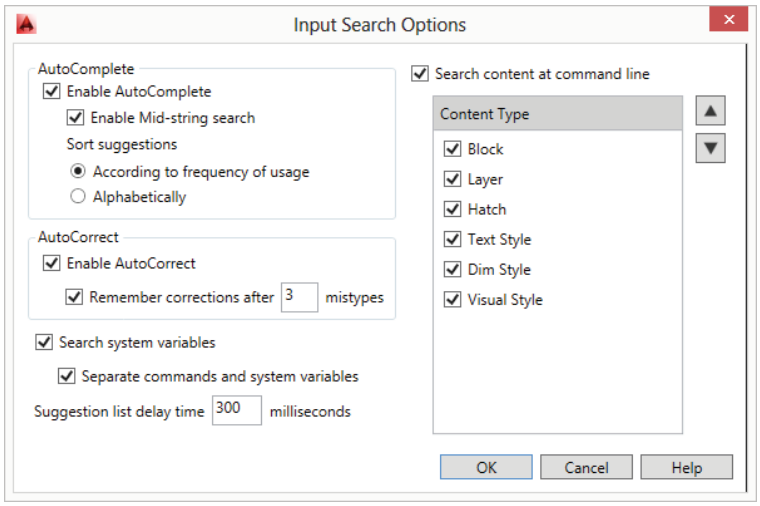


FIGURE 4.5 Customizing the input search options for the Command prompt

Command Input History

As input is entered at the command prompt, it is stored and can be recalled later. When a command is not active, you can access recently used commands; when a command is active, you can

access recently entered coordinate input. The `inputhistorymode` system variable controls the type of input that is recorded and where it can be accessed from. Recent input can be accessed from the command line on both Windows and Mac OS (see Figure 4.6). On Windows systems, you can also access recent input from the drawing-window shortcut menu. Additionally, you can press the up- and down-arrow keys on the keyboard to cycle through recent commands and input.

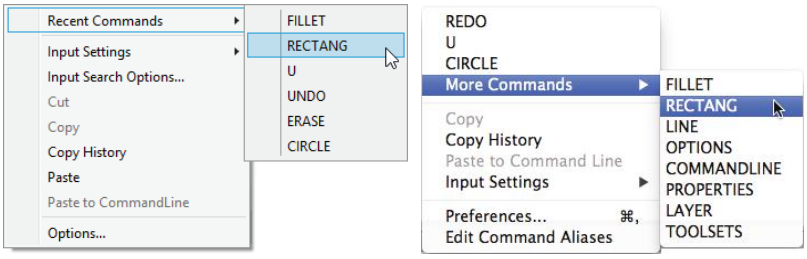


FIGURE 4.6 Accessing recently used commands and coordinate values

You can also use the `cmdinputhistorymax` system variable to control the number of coordinate input entries that are recorded. If you reduce the value of the `cmdinputhistorymax` system variable, the changes might not be obvious until you restart the application, reopen any drawings that are currently open, or open another drawing.

Dynamic Input

Dynamic input displays prompts, tooltips, and pointer and dimension input fields at the cursor when the feature is enabled. The Dynamic Input tab of the Drafting Settings dialog box (`dsettings` command), as shown in Figure 4.7, allows you to customize the fields and tooltips used for dynamic input. You can also use the system variable `dynmode` to enable or disable dynamic input. On Windows, you can also control the display of the Z field of the pointer input field on the 3D Modeling tab in the Options dialog box (`options` command).

The following explains what each area of the Dynamic Input tab controls:

Pointer Input Clear Enable Pointer Input to suppress the ability to enter coordinate values when prompted for a point by the current command. Click Settings to display the Pointer Input Settings dialog box and specify the coordinate format that should be applied when a command requests a second or next point. You can also control when the coordinate tooltips are displayed. The `dynpicoords` and `dynpiformat` system variables can be used to control the pointer input options at the command prompt.

Dimension Input Clear Enable Dimension Input Where Possible to suppress the ability to enter distance and angular values based on the previous point when prompted for a second or next point by the current command or while using grips. Click Settings to display the Dimension Input Settings dialog box and control the number and type of dimension input fields to display when being requested for coordinate, distance, and angular values while a command is active. The `dyndivis` and `dyndigrip` system variables can be used to control the dimension input options at the command prompt.

Dynamic Prompts The settings in this section control if the prompts and messages displayed near the crosshairs should be merged into a single tooltip or displayed as multiple tooltips. The `dynprompt`, `dyninfotips`, `dyntooltips`, and `tooltipmerge` system variables can be used to control the appearance of the dynamic input tooltips at the command prompt.

Drafting Tooltip Appearance Clicking this button displays the Tooltip Appearance dialog box, which allows you to control the color, size, and transparency for dynamic input tooltips and fields.

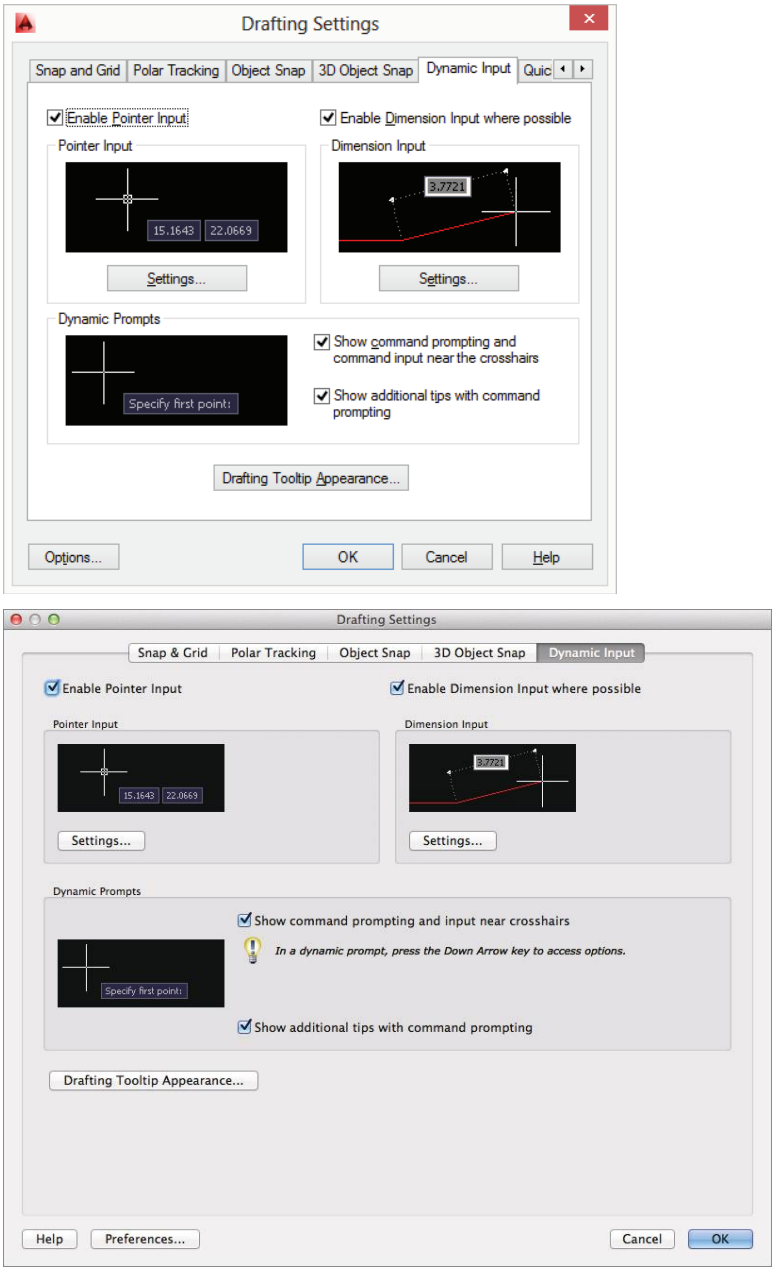


FIGURE 4.7 Controlling the behavior and appearance of dynamic input

Creating and Modifying Command Aliases

Command aliases allow you to quickly start a command or access a system variable by entering a few letters at the command prompt instead of entering its full name or locating it in the user interface. Don't confuse command aliases with keyboard shortcuts, which are key combinations that require you to press a combination of the Ctrl/Control, Shift, Alt/Option, or Command keys with a letter, number, or virtual key. I discuss more about creating custom shortcut keys in AutoCAD on Windows in Chapter 5, "Customizing the AutoCAD User Interface for Windows."

AutoCAD comes with hundreds of predefined command aliases that are available for many of the most commonly used commands and system variables. You can save time performing everyday drafting tasks by learning the command aliases for the commands you use most frequently, since you do not need to move the cursor outside the drawing area to start a command.

Some of the commonly used command aliases are `l` for `line`, `e` for `erase`, and `dli` for `dimlinear`. If a command alias does not exist for the commands or system variables you commonly use, you can define new or override existing aliases by editing the `acad.pgp` file.

In addition to starting an AutoCAD command, command aliases can be used to start external applications from AutoCAD on Windows only. For example, you can launch Windows Explorer or File Explorer by using the alias `explorer` or start Notepad by entering `notepad` at the AutoCAD command prompt.

TIP As an alternative to using external commands, you can define a command that uses the AutoLISP `startapp` function to call an external application from AutoCAD on Windows or Mac OS. For information on AutoLISP and the `startapp` function, see the AutoCAD Help system.

Defining Command Aliases

Command aliases are stored in the `acad.pgp` file, which is a plain ASCII text file that can be edited using Notepad on Windows or TextEdit on Mac OS. The `acad.pgp` file is divided into two sections: external commands and command aliases. I cover external commands in the section "Defining External Commands (Windows Only)" later in this chapter.

Two pieces of information are required to define a command alias: an abbreviation, which is the shortened command name (or alias) you want to use to start a command, and then the name of the AutoCAD command you want to start. A command alias can only start a command; it can't pass any values to the options of the command it starts. If you want to start a command and pass values to the options of the command that is started, you can use a script, action macro, or a custom command defined with AutoLISP. I discuss scripts and action macros in Chapter 8, "Automating Repetitive Tasks."

A command alias has the following syntax:

```
<abbreviation>, *<command>
```

Table 4.2 lists some of the most commonly used command aliases and how they are defined in the `acad.pgp` file.

TABLE 4.2: Commonly used command aliases

ABBREVIATION	COMMAND
E,	*ERASE
C,	*CIRCLE
CO,	*COPY
DLI,	*DIMLINEAR
L,	*LINE
Z,	*ZOOM

The `acad.pgp` file is loaded each time AutoCAD is started, when the `reinit` command is used, or when the `re-init` system variable is set to a value of 16. When the `acad.pgp` file is loaded, AutoCAD reads the file in a top-down order looking for command alias definitions. If AutoCAD encounters one or more command aliases with the same abbreviations, it remembers only the last command alias it read as it follows the "last one in wins" rule.

Since the `acad.pgp` file allows you to define more than one command alias with the same abbreviation and AutoCAD uses the "last one in wins" rule when loading command aliases, Autodesk recommends that you place all your custom command aliases under the User Defined Command Aliases section of the `acad.pgp` file. The User Defined Command Aliases section is located at the bottom of the `acad.pgp` file.

The following example explains how to define a new command alias and redefine two of the standard command aliases that come with AutoCAD:

1. Do one of the following:
   - On the ribbon, click Manage tab ➢ Customization panel ➢ Edit Aliases drop-down menu ➢ Edit Aliases (Windows).
   - At the command prompt, type **`ai_editcustfile`** and press Enter. At the `Custom File to edit:` prompt, type **`acad.pgp`** and press Enter (Windows).
   - Click Tools menu ➢ Customize ➢ Edit Command Aliases (PGP) (Mac OS).
   - At the command prompt, type **`aliasedit`** and press Enter (Mac OS).
2. In Notepad or TextEdit, scroll to the bottom of the PGP file and enter the following text:

```
REV, *REVCLOUD
L, *PLINE
JU, *JUSTIFYTEXT
```

3. Click File ➢ Save to save the changes to the PGP file.
4. In AutoCAD, at the command prompt, type **`re-init`** and press Enter. At the `Enter new value for RE-INIT <0>:` prompt, enter **`16`** and press Enter.

5. Try the command aliases you entered in step 2 to make sure they work as expected. Normally `L` would start the `line` command, but here you defined a new command alias that starts the `pline` command instead of `line`. If you need to make any changes, switch back to the text editor with the PGP file and make any necessary changes. Then reload the PGP file as explained in step 4.

TIP In AutoCAD on Windows, you can use the Express Tools Command Alias Editor (`aliasedit` command) to define and modify both command aliases and external commands in the `acad.pgp` file.

Defining External Commands (Windows Only)

External commands are stored in the `acad.pgp` file, just like command aliases are, but they are only supported in AutoCAD on Windows.

External commands use the following syntax:

```
<abbreviation>,[<shell_call>],<flag>,[*]<prompt>,
```

abbreviation The name that you enter at the command prompt to start the external application.

shell_call Specifies which command to use from the MS-DOS or Windows command prompt. Some examples of commands that you can use are `START`, which allows you to launch an application, `DEL` to delete a file, or `TYPE` to display the contents of a file.

flag Controls how the application is called. The available values can be found in the `acad.pgp` file by searching on the text `External command format`.

prompt Specifies an instructional message to the user at the AutoCAD command prompt, and requests input from the user that can be passed to the command in the *shell_call* argument.

The following shows a couple of the external commands that are predefined in the `acad.pgp` file:

```
DEL,       DEL,           8,File to delete: ,
NOTEPAD,   START NOTEPAD,  1,*File to edit: ,
```

The following steps explain how to define a new external command to start the Windows Calculator (`calc.exe`):

1. Do one of the following:
   - On the ribbon, click Manage tab ➢ Customization panel ➢ Edit Aliases drop-down menu ➢ Edit Aliases.
   - At the command prompt, type **`ai_editcustfile`** and press Enter. At the `Custom File to edit:` prompt, type **`acad.pgp`** and press Enter.
2. In Notepad, scroll to the bottom of the PGP file and enter the following text:

   ```
   CALC, START CALC, 1,,
   ```

3. Click File ➢ Save to save the changes to the PGP file.
4. In AutoCAD, at the command prompt, type **re-init** and press Enter. At the `Enter new value for RE-INIT <0>:` prompt, type **16** and press Enter.
5. At the command prompt, type **calc** and press Enter. If successful, the Windows Calculator should be displayed. If not, switch back to Notepad and make sure you correctly defined the external command and that you reloaded the PGP file with the `re-init` system variable.

Adding Synonym and AutoCorrect Entries (Windows Only)

Starting with AutoCAD 2014, two new command alias–related features were introduced. These features are known as synonyms and AutoCorrect, and they are available in AutoCAD on Windows only. Synonym and AutoCorrect entries are stored in the `acadSynonymsGlobalDB.pgp` and `AutoCorrectUserDB.pgp` files, respectively, and follow the same syntax as the command aliases I mentioned in the section "Defining Command Aliases" earlier in this chapter.

NOTE The synonym and AutoCorrect features are available in only AutoCAD 2014 and AutoCAD 2014–based products.

Synonym entries allow you to provide a natural name or term as an alternative to an AutoCAD command; this should not be a shortened name, though, like the command aliases in the `acad.pgp` file. For example, you might commonly use the term *symbol* at your company or in your industry as opposed to *block*. AutoCAD does not have a `symbol` command, so in previous releases you had to learn both the AutoCAD lingo and its commands. Now with synonyms, it is much easier to enter some industry terms like *symbol* or *callout* at the AutoCAD command prompt and start an associated command. You can add your own or change existing synonyms in the `acadSynonymsGlobalDB.pgp` file using Notepad.

The following example explains how to add a synonym entry for the term *bevel* and have it start the `chamfer` command. For more information on defining command aliases and reloading PGP files, see the "Defining Command Aliases" section.

1. On the ribbon, click Manage tab ➢ Customization panel ➢ Edit Aliases drop-down menu ➢ Edit Synonym List.
2. In Notepad, scroll to the bottom of the PGP file and enter the following text:
   `BEVEL, *CHAMFER`
3. Save the changes to the PGP file, and then reload it with the `re-init` system variable.
4. At the AutoCAD command prompt, type **bevel** and press Enter. If successful, the `chamfer` command should be started.

AutoCorrect entries share the same syntax as synonyms and command aliases, but you typically do not add new entries to the `AutoCorrectUserDB.pgp` file directly using Notepad. Instead, AutoCorrect entries are added automatically as a result of incorrectly entering the name of a command or system variable and then choosing the correct spelling from the suggestions list.

You must misspell a name and select the correct one three times before the term will be added to the `AutoCorrectUserDB.pgp` file. You enable the AutoCorrect feature and the number

of times a correction must be made before an entry is added to the PGP file by using the Input Search Options dialog box (`inputsearchoptions` command); see Figure 4.5 earlier.

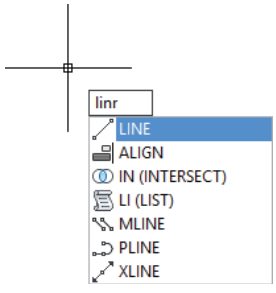
The following steps explain how to check if AutoCorrect is enabled, and if it isn't, how to enable it:

1. Right-click over the command-line window and click Input Search Options.
2. In the Input Search Options dialog box, make sure Enable AutoCorrect and Remember Corrections After Mistypes are both checked. Set Remember Corrections After Mistypes to a value of 3. Click OK.

You can add an AutoCorrect entry from the AutoCAD user interface for a misspelling of the `line` command by performing the following steps:

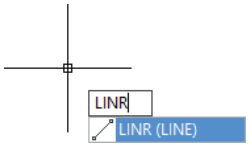
1. At the command prompt, type **`linr`** and wait for the suggestions list to open.
2. In the suggestions list (see Figure 4.8) that appears, click LINE.

FIGURE 4.8
Command suggestion list based on typed characters



3. Press Esc to end the `line` command.
4. Repeat steps 1–3 two more times.
5. At the command prompt, type **`linr`**. This time you should see a new AutoCorrect entry in the suggestion list (see Figure 4.9). Pressing Enter starts the `line` command. Press Esc to end the `line` command.

FIGURE 4.9
AutoCorrect entry in the command suggestion list



Working with System and Environment Variables

Commands and dialog boxes are used to create, modify, and manage drawing objects and application settings. When most commands or dialog boxes are used, they store and access settings that might be held temporally in memory or persist within individual drawing files across

application sessions. Some of these settings are exposed for users and developers to gain a level of control over AutoCAD; the exposed settings are known as *system and environment variables*.

System and environment variables can be accessed from the command prompt or through supported programming languages. Of the two variable types, system variables are the most common. Both types of variables can store values that represent a string, number, coordinate point, lists of values, and much more. You can query and set the value of a system variable with the `setvar` command, or with the `getvar` and `setvar` AutoLISP functions. Environment variables can be queried and set using the `getenv` and `setenv` AutoLISP functions. There is no equivalent to the `setvar` command that allows you to directly access and set the value of an environment variable.

You can learn more about accessing system and environment variables with AutoLISP and VBA from the AutoCAD Help system. If you are using Managed .NET and C++ with ObjectARX®, you will want to use the documentation that comes with the ObjectARX SDK to learn how to access system and environment variables.

NOTE You can learn more about the AutoCAD Managed .NET and ObjectARX programming interfaces at `www.objectarx.com`.

Listing and Editing System Variables

The `setvar` command allows you to list all supported system variables and their current values. The following explains how to obtain a listing of all supported system variables:

1. At the command prompt, type **`setvar`** and press Enter.
2. At the `Enter variable name or [?]:` prompt, type **`?`** and press Enter.
3. At the `Enter variable(s) to list <*>:` prompt, type **`*`** and press Enter. The command displays about 20 entries and waits for you to press Enter to continue. This allows you time to review the values of the system variables that are listed.

TIP You can use wildcard matching to display a refined system-variable listing. For example, you can enter **`*DIM*`** at the `Enter variable(s) to list <*>:` prompt to list all the system variables that start with or contain the characters `DIM`.

4. Press Enter when you are ready to review the next entries. Continue pressing Enter to step through the system variables or press Esc to cancel the `setvar` command.

The system-variable list displayed by the `setvar` command contains the name, current value, and read-only state of each variable listed. You can access additional information about a system variable from the product's Help system.

```
3DCONVERSIONMODE  1
3DDWFPREC         2
```

```
APBOX            0
APERTURE         10
Press ENTER to continue:
```

TIP You can export a list of all system variables and their values by turning command logging on before starting the `setvar` command. Command logging can be turned on using the `logfileon` command, and then turned off again using the `logfileoff` command. The filename and folder where the log file is created is stored in the `logfilename` and `logfilepath` system variables.

Use the following steps to access help for a system variable (or command) from the command prompt. These specific steps explain how to access help for the `dimtp` system variable:

1. At the command prompt, type **`dimtp`**. Do not press Enter as you normally would.
2. Press F1 (Windows) or Control+F1 (Mac OS). The Help window opens to the topic related to the system variable.

TIP On Windows, you can also use the Express Tools System Variable Editor (`sysvdlg` command) to access a list of all available system variables; their current value and the values that can be expected or applied are also listed.

You can change the value of a system variable by entering its name directly at the command prompt or at the `Enter variable name or [?]:` prompt of the `setvar` command. Follow these steps to change the value of the `dynmode` system variable, which enables the use of dynamic input:

1. At the command prompt, type **`dynmode`** and press Enter.
2. At the `Enter new value for DYNMODE <3>:` prompt, enter **`0`** and press Enter. Record the value in the angle brackets (< >). The Dynamic Input icon on the status bar reflects that the feature is now disabled. You could also verify that it is displayed by starting a command, such as `line` or `circle`.
3. At the command prompt, type **`dynmode`** and press Enter. Type **`3`** or the value that was in the angle brackets, and press Enter to restore the default or previous value.

When you assign a value to a system variable, the value you are assigning must be of the correct data type. For example, the `dynmode` system variable expects an integer and if passed a string of `"3"` instead of `3`, an error message is displayed and the command ends. The same happens if you enter an unexpected value. Refer to the Help topic that covers the system variable for information about the type of data and the values that the system variable expects.

Listing and Editing Environment Variables

Environment variables are kind of a mystery at times since they are not really documented, so in many ways they must be discovered. A few of them can be found in the AutoCAD product documentation, but the best option is to use your favorite search engine and see what other

users have found. I do not have a great understanding as to why they are not documented like system variables.

Even though they are not commonly used, there are three environment variables you might need to change or query from time to time based on your drawing requirements:

MaxArray Specifies the maximum number of objects that can be created during a single use of the ARRAY command.

MaxHatch Specifies the maximum number of objects that can be displayed within a single hatch object. If you see the message `Hatch Pattern Too Dense`, you will need to increase the number of objects that can be displayed for hatch objects. If the limit is exceeded, the hatch object appears as a single fill. You might need to close and reopen a drawing for the new setting to be applied.

ACAD Specifies the paths listed under the Support File Search Path of the Files tab in the Options dialog box on Windows or Application tab in the Application Preferences dialog box on Mac OS.

When you want to query or set the value of an environment variable, you need to use the `getenv` and `setenv` AutoLISP functions. The name of the variable is case specific, so `MaxArray` —not `MAXARRAY` or `maxarray`— must be provided to the functions. Also, when setting the value of an environment variable you must provide it as a string. In the following steps, you return and set the value of the `MaxHatch` environment variable:

1. At the command prompt, enter

   ```
   (setq mh (getenv "MaxHatch"))
   ```

   When you press Enter, the value of the environment variable is returned as a string to the command-line window.

2. At the Command prompt, enter

   ```
   (setenv "MaxHatch" "1000")
   ```

   When you press Enter, the new value is repeated in the command-line window.

3. At the Command prompt, enter

   ```
   (setenv "MaxHatch" mh)
   ```

   When you press Enter, the previous value is restored.

Chapter 5

Customizing the AutoCAD User Interface for Windows

The Autodesk® AutoCAD® user interface provides you with one of the best opportunities to increase your productivity without learning a programming language. Many of the user-interface elements that allow you to start a command or toggle a system variable with a click of a button or the press of a key can be customized. By customizing the user interface, you can reorganize it to better fit the way you work, add the commands that you frequently use, and even remove those commands that you do not use.

You customize the user interface through a combination of direct manipulation and the Customize User Interface Editor—or CUI Editor, as it is commonly known. Direct manipulation can make it fast and easy to reorganize elements, but there are limitations, as not all elements are supported and new user-interface elements can't be added. The CUI Editor provides the most control over the elements of the user interface.

As you make changes to the user interface, the changes are stored as part of the Windows Registry or in the main customization (CUIx) file. CUIx files contain the definitions for many of the elements that are displayed in the AutoCAD application window, such as the buttons on a ribbon panel, items on the menu bar, or even shortcut key combinations.

Getting Started with the CUI Editor

The CUI Editor is the tool that you will need to become familiar with to customize the AutoCAD user interface. Figure 5.1 shows the CUI Editor and highlights some of the areas that I will discuss. The editor might appear a bit overwhelming at first—and to be honest it can be because of everything it does—but there is nothing to be afraid of; I will guide you one step at a time. The CUI Editor is a much better solution for those new to customizing the AutoCAD user interface than what was available in the releases prior to AutoCAD 2006. In those versions, a dialog box offered only limited customization options. You needed to use an ASCII text editor like Notepad to customize all available elements.

Not that I have anything against Notepad; it just does not know what needs to be done to create a toolbar, add a toolbar button, or define a pull-down menu with a series of items. If you forget a character, attempting to load the miscoded menu/customization file often results in a cryptic message from AutoCAD. The CUI Editor lowers the learning curve for creating and modifying user-interface elements.

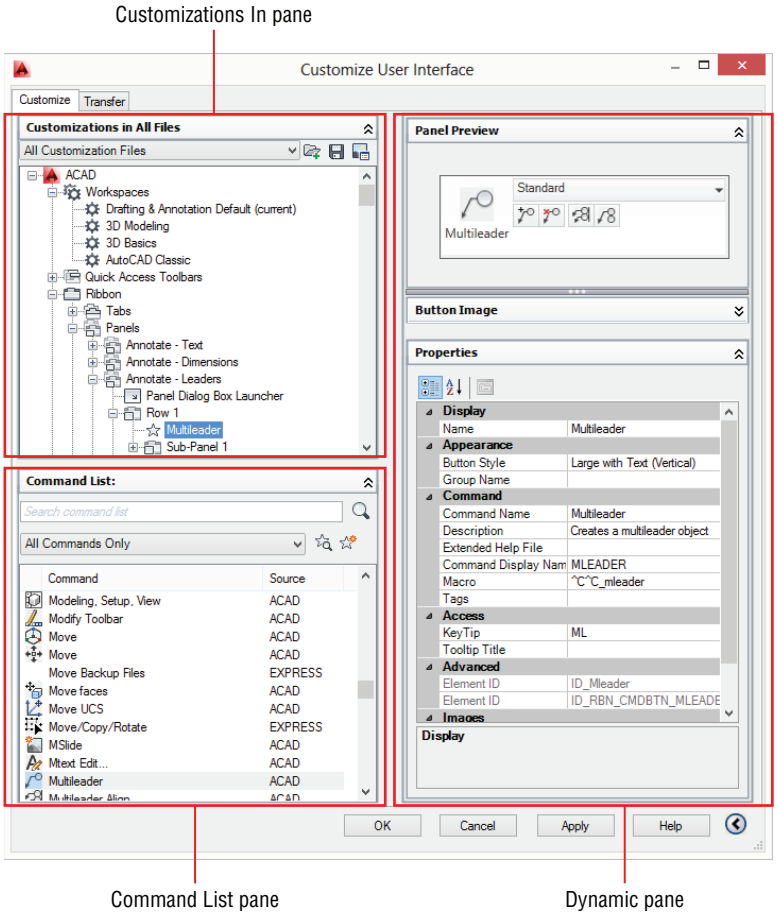


FIGURE 5.1
The CUI Editor lets you create and modify user-interface elements.

You can display the CUI Editor using one of the following methods:

- On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
- Right-click a button on the Quick Access toolbar or a standard toolbar and click Customize Quick Access Toolbar or Customize, respectively.
- At the command prompt, type **`cui`** and press Enter.

When the CUI Editor is displayed, notice that there are two tabs. Each tab is divided into areas called *panes*. The two tabs in the CUI Editor and their purpose are as follows:

Customize Tab Use the Customize tab to create, modify, and organize the user-interface elements that come with AutoCAD or those that you create. You will also use this tab to create workspaces that allow you to control when and where specific user-interface elements are displayed. This tab is divided into three panes:

Customizations In Pane Here you will find a listing of the CUIx files that are currently loaded and the user-interface elements that they contain. When a user-interface element

is expanded, you see each of the items that make up that particular element, such as the buttons on a ribbon panel or the items on a pull-down menu. When you select a user-interface element, command, or control in this pane, its properties can be changed in the Dynamic pane.

Command List Pane Here you'll find a list of the commands and controls that you can add to the user-interface elements in the Customizations In pane. New custom commands for use in the user-interface are created in this pane. Selecting a command from this pane displays its properties in the Dynamic pane, where you can change the image, name, macro, and other settings that define how a command appears in the user interface.

Dynamic Pane Displays the properties of an item selected from either the Customizations In or Command List pane. Based on the item selected, one or more of eight different subpanes could be displayed. I cover each of these panes later in this chapter as I explain how to customize the elements of the user interface.

Transfer Tab Use the Transfer tab to copy user-interface elements between customization (CUIx) files, migrate user-interface elements from an earlier release, and create new or save existing CUIx files.

TIP You can resize a pane by positioning the cursor between two panes and dragging when the cursor changes to two arrows that point in opposite directions. The panes on the Customize tab can also be collapsed by clicking a pane's title bar.

Creating Commands and Defining Command Macros

Commands are the primary component of elements in the AutoCAD user interface, and they are created in the Command List pane of the CUI Editor. The properties of a command in a CUIx file define the sequence of AutoCAD commands and options that will be executed when the command is used, and how the command should appear on a user-interface element. The sequence of AutoCAD commands and options that are assigned to a command are contained in a *macro*. The macro is the most significant property of a command in a CUIx file.

Understanding the Basics of a Command Macro

A macro defines the input that is sent to the AutoCAD command prompt when a user-interface element is used; it can, but does not necessarily need to, start and complete a command in a single command macro. You could start a command with one macro and then click another button to send an expected value to the active command. An example might be where one macro starts a custom AutoLISP® routine that prompts you for a bolt or window size, and rather than typing a size each time, you use a second macro to pass a value to the routine.

For the most part, a macro is similar to the input that you enter at the command prompt to start and complete an AutoCAD command, but it can also contain special characters that control its execution. For example, the following might be what you normally would do to draw a circle with a diameter of one-eighth of an inch:

1. At the command prompt, type **`circle`** and press Enter.
2. At the `Specify center point for circle or [3P/2P/Ttr (tan tan radius)]:` prompt, specify a point with the input device or type a value at the command prompt.

3. At the `Specify radius of circle or [Diameter] <0.1875>:` prompt, type **d** and press Enter.
4. At the `Specify diameter of circle <0.3750>:` prompt, type 0.125 and press Enter.

An example of a command macro using the same input as the previous example might look something like this:

```
^C^C._circle;\_d;0.125;
```

As you can see, I used five special characters in the example macro that were not present as part of the original input entered at the command prompt: `^` (caret), `.` (period), `_` (underscore), `;` (semicolon), and `\` (backslash). Table 5.1 explains the significance of each macro component.

TABLE 5.1: Meaning of macro components

Macro Component	Description
`^C^C`	Simulates the pressing of the Esc key twice.
`._circle`	Passes the `circle` command to the command prompt.
`;`	Simulates the pressing of the Enter key to start the `circle` command.
`\`	Pauses for the user to specify the center point of the circle.
`_d`	Indicates that the Diameter option of the `circle` command will be used.
`;`	Simulates the pressing of the Enter key to accept the Diameter option.
`0.125`	Specifies the value for the Diameter option.
`;`	Simulates the pressing of the Enter key to accept the diameter value. Since this is the last expected value, the `circle` command ends too.

Table 5.2 lists the most common special characters used in a command macro.

TABLE 5.2: Special characters that can be used in macros

Special Character	Description
`^C`	Equivalent to pressing Esc.
`;`	Equivalent to pressing Enter.
`[blank space]`	Equivalent to pressing Enter or spacebar based on the expected input of the current prompt.

TABLE 5.2: Special characters that can be used in macros *(CONTINUED)*

SPECIAL CHARACTER	DESCRIPTION
\	Allows the user to provide input.
.	Instructs AutoCAD to use the command's standard definition even when a command might have been undefined with the `undefine` command.
_	Instructs AutoCAD to use the global command name or option value instead of the local name or value provided. This allows the macro to function as expected when used with a different language of the AutoCAD release.
*	Repeats the AutoCAD command after the asterisk character until the user cancels the command. Example macro from the Point, Multiple Point command in the `acad.cuix` file: `*^C^C_point`
$M=	Indicates the start of a DIESEL expression. Example expression from the UCS Icon, On command in the `acad.cuix` file: `$M=$(if,$(and,$(getvar,ucsicon),1),^C^C_ucsicon _off,^C^C_ucsicon _on)`

TIP To learn about other special and control characters that can be used in a command macro, search the AutoCAD Help system using the keywords *characters in macros*.

When combining multiple commands into a single menu macro, you will want to first step through the sequence at the command prompt. Doing this can help you identify which commands, options, and values you want to use. The following example demonstrates the commands and options you might use to create and set as current a layer named Notes and then draw a multiline text object:

1. At the command prompt, type **`-layer`** and press Enter.
2. At the `Enter an option` prompt, type **`m`** and press Enter.
3. At the `Enter name for new layer (becomes the current layer) <0>:` prompt, type **`Notes`** and press Enter.
4. At the `Ente0r an option` prompt, type **`c`** and press Enter.
5. At the `New color [Truecolor/COlorbook]:` prompt, type **`8`** and press Enter.
6. At the `Enter name list of layer(s) for color 8 <Notes>:` prompt, press Enter.
7. At the `Enter an option` prompt, press Enter.
8. At the command prompt, type **`-mtext`** and press Enter.
9. At the `Specify first corner:` prompt, specify a point in the drawing area.

10. At the `Specify opposite corner or [Height/Justify/Line spacing/Rotation/Style/Width/Columns]:` prompt, type **`j`** and press Enter.
11. At the `Enter justification [TL/TC/TR/ML/MC/MR/BL/BC/BR] <TL>:` prompt, type **`tl`** and press Enter.
12. At the `Specify opposite corner or [Height/Justify/Line spacing/Rotation/Style/Width/Columns]:` prompt, type **`h`** and press Enter.
13. At the `Specify height <0.2000>:` prompt, type **`0.25`** and press Enter.
14. At the `Specify opposite corner or [Height/Justify/Line spacing/Rotation/Style/Width/Columns]:` prompt, type **`r`** and press Enter.
15. At the `Specify rotation angle <0>:` prompt, type **`0`** and press Enter.
16. At the `Specify opposite corner or [Height/Justify/Line spacing/Rotation/Style/Width/Columns]:` prompt, type **`w`** and press Enter.
17. At the `Specify width:` prompt, type **`7.5`** and press Enter.
18. At the `MText:` prompt, type **`NOTE: ADA requires a minimum turn radius of`** and press Enter.
19. At the `MText:` prompt, type **`60" (1525mm) for wheelchairs.`** and press Enter.
20. Press Enter again to end the `mtext` command and leave the command-line window open.

What's That Hyphen?

As I discussed earlier, commands that display dialog boxes or palettes should be avoided in macros when you want to use specific values. Adding a leading hyphen to many commands that normally display a dialog box or palette starts an alternate command that displays a series of prompts instead. For example, use `-layer` instead of `layer` when you want to create a layer from a command macro, or `-insert` instead of `insert` to insert a block. See Chapter 8, "Automating Repetitive Tasks," for a listing of alternative commands and system variables that allow you to avoid opening dialog boxes and palettes.

After you've worked through the process at the command prompt, you can use that information to convert the process to a macro. The next steps walk you through the process of converting input entered at the command prompt into a command macro:

1. Press F2 to expand the command-line history or display the AutoCAD Text Window.
2. In the History area, select the command prompts that were displayed and input that you entered (see Figure 5.2). Right-click and click Copy.
3. At the command prompt, enter **`notepad`** and press Enter twice to launch Notepad.
4. In Notepad, click in the editor window and press Ctrl+V to paste the copied text from the command-line history.

FIGURE 5.2 Command-line history of the input you previously entered

5. From the pasted text, remove the two informational lines `Current layer:` and `Current text style:` and their values.
6. Replace the `Specify first corner:` prompt with a **** (single backslash character).
7. Remove all the other prompts before the input you entered.
8. After each line, add a ; (semicolon), with the exception of the line that contains the backslash. After this and the previous three steps you should have the following left in Notepad:

```
-layer;
m;
Notes;
c;
8;
;
;
-mtext;
\
j;
tl;
h;
0.25;
r;
0;
w;
7.5;
NOTE: ADA requires a minimum turn radius of;
60" (1525mm) diameter for wheelchairs.;
;
```

9. Enter **^C^C** before the first line to make sure no other command is active when the macro is used.

10. Place the cursor at the end of each line and press Delete to move all input to a single line. Your finished macro should look like this:

```
^C^C-layer;m;Notes;c;8;;;-mtext;\j;tl;h;0.25;r;0;w;7.5;
   NOTE: ADA requires a minimum turn radius of;60" (1525mm) for wheelchairs.;;
```

11. Click File ➢ Save As.
12. In the Save As dialog box, browse to the `MyCustomFiles` folder that you created under the `Documents` (or `My Documents`) folder, or the location where you want to store the text file.
13. In the File Name text box, type **`mynotemacro.txt`** and click Save.
14. Do not close Notepad; you will use the macro you just created in the next section, when you create a command for use in the user interface.

TIP Add the text **`._`** (period underscore) in front of each command name and an **`_`** (underscore) in front of the values that represent an option name. This ensures that your macro works correctly if a command is undefined or the command macro is used on a non-English AutoCAD release.

Here's how the macro you just created would look after prefixing commands with `._` and options with `_`:

```
^C^C._-layer;_m;Notes;_c;8;;;._-mtext;\_j;_tl;_h;0.25;_r;0;_w;7.5;
   NOTE: ADA requires a minimum turn radius of;60" (1525mm) for wheelchairs.;;
```

Creating and Modifying Commands

Before you can use your macro, you first need to learn how to create a new command in a CUIx file. Commands are created under the Command List pane of the CUI Editor. You can also locate and modify existing commands in the CUIx files that are currently loaded.

The following example explains how to create a command for the macro that you created in the previous section in a current CUIx file. If you did not complete the steps for the previous example, you can open the `NoteMacro.txt` exercise file that is available for download from `www.sybex.com/go/autocadcustomization`. If you did complete the previous example but closed Notepad, launch Notepad and open the file `MyNoteMacro.txt` from the `MyCustomFiles` subfolder under the `Documents` (or `My Documents`) folder, or the location you used.

1. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface (or at the command prompt, type **`cui`** and press Enter).

2. In the CUI Editor, from the Command List pane select Create A New Command.

NOTE When you create a new command, it is added to the customization (CUIx) file that is selected from the drop-down list at the top of the Customizations In pane. If you want to add a command to a partial customization file, make sure it is selected before creating the command. I discuss the types of customization files later in this chapter in the "Working with Customization Files" section.

3. In the Properties pane (see Figure 5.3), type **`Wheelchair Note`** in the Name field. The Name field is used to identify the command in the Command List pane and is part of the

tooltip (shown in Figure 5.12, later in this chapter) that is displayed when the cursor hovers over the command in the user interface.

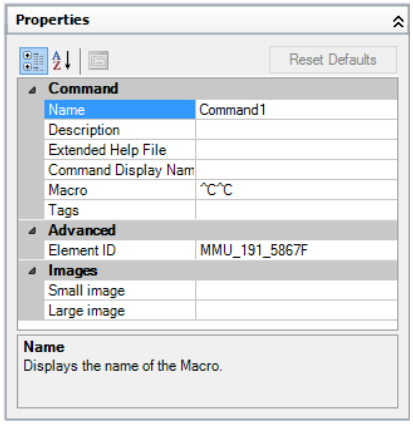


FIGURE 5.3 Defining the properties of a command

4. Click in the Macro field and then click the ellipsis […] button. In the Long String Editor, clear the current value in the text box. Enter the macro that you created in the previous exercise or copy/paste the contents of the `MyNoteMacro.txt` or `NoteMacro.txt` file into the text box. Click OK. The macro defines the actions that AutoCAD will perform when the command is used from the user interface.
5. Optionally, in the Description field enter **`ADA Wheelchair minimum radius note`**. The text entered in this field helps to describe what the command is used for and is part of the tooltip that is displayed when the cursor hovers over the command in the user interface.
6. Optionally in the Extended Help File field, specify an XAML file that contains additional text and images that describe what the command does. I do not cover creating extended help files in this book; search on the keywords *extended help* in the AutoCAD Help system to learn more.
7. Optionally in the Command Display Name field, enter **`-LAYER, TEXT`**. The text entered in this field helps the user to identify which AutoCAD commands are used as part of the macro, and is part of the tooltip displayed when the cursor is over the command in the user interface.
8. Optionally, click in the Tags field and then click the ellipsis […] button. In the Tag Editor, click in the Tags text box and enter **`Wheelchair,Note`**. Click OK.

NOTE Tags make it easier to locate a command without looking for it in the user interface. You can search for a command that is assigned a tag using the Search field of the Application menu; the Search field is accessed by clicking the Application button located near the upper-left corner of the AutoCAD application window.

9. Click Apply to save the changes you made to the properties of the new command (see Figure 5.4).

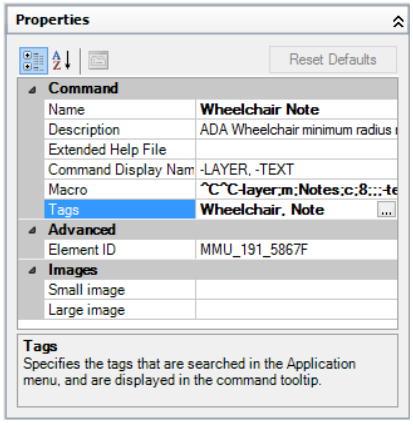


FIGURE 5.4
The properties of the completed command

A command can be edited by selecting it from the Command List pane and then making changes to it in the Properties pane, just as you did when you created the new command.

TIP Click the Filter The Command List By Category drop-down list under the Command List pane and select Custom Commands to list only the custom commands that have been added to any of the loaded CUIx files.

Creating and Assigning an Image to a Command

While using images with your commands is optional, you should consider adding an image to each of the commands that you create in a CUIx file to provide the most flexibility. Although not all user-interface elements display an image, most of the common user-interface elements do. AutoCAD provides a basic image editor that allows you to create images for your commands right inside the CUI Editor. However, if you or someone else in your company has experience with a different image editor, you can use that software.

Here are the basic requirements your images need to meet:

- Small images should be 16×16 pixels in size.
- Large images should be 32×32 pixels in size.
- Images need to be in the BMP file format.

If an image is created using the Button Editor inside the CUI Editor, that image is saved as part of the CUIx file. You can export an image file if you want to edit the image outside of AutoCAD. You also can import into a CUIx file images that you created or edited outside of AutoCAD and then assign those images to a command. An alternative to importing images into the CUIx file is to create a resource DLL file that has the same name as the CUIx file being loaded into AutoCAD. This method is more common with third-party utilities that use CUIx files for their user-interface elements.

The following example explains how to create a custom image for the Wheelchair Note command you created in the previous section:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the CUI Editor, from the Command List pane select the Wheelchair Note command.

3. Under the Button Image pane, select one of the images from the Image list. It does not matter which image you select unless there is an image that is similar to the image you want to create. If there is a similar image, select it.
4. Under the Apply To section, select Both.
5. Click Edit.
6. In the Button Editor (see Figure 5.5), click Clear.

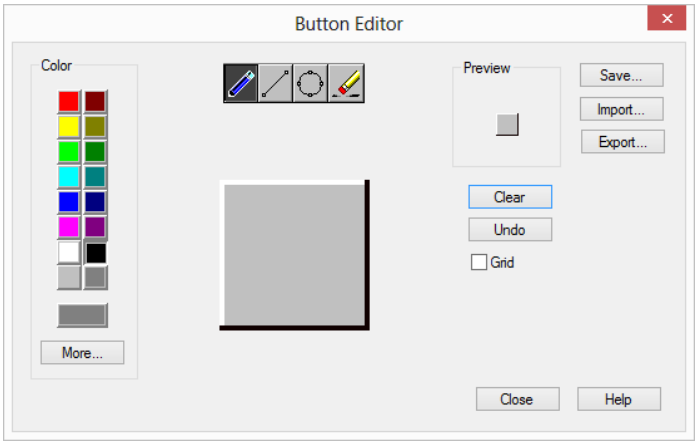


FIGURE 5.5
Creating a custom image

7. Click the Grid check box to display a grid of pixel squares over the image canvas.
8. Click the Pencil drawing tool located above the image canvas to edit the image.
9. Click one of the color swatches from the left side or click More to display the Select Color dialog box. If you click More, choose a color that is different from the standard colors.
10. Click (or drag over) the image canvas to create your image. Draw an image that you feel conveys the idea of a Wheelchair Note. Figure 5.6 shows an example of an image that I created. It can be found in the files available for download from this book's web page; the file is named `WheelchairNote.bmp`.

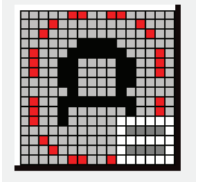


FIGURE 5.6
Example custom image

11. After you have created your image, click Save.
12. In the Save Image dialog box, in the Image Name text box enter **`WheelchairNote`**. Click OK.
13. Click Close to return to the main dialog of the CUI Editor. The image you created should now be assigned to both the Small Image and Large Image fields of the command.

14. Click Apply to save the changes to the command.

If you have an externally saved file that you want to use for a command, you can do the following:

1. In the CUI Editor, from the Command List pane select the command that you want to assign a button image to.
2. In the Button Image pane, right-click the Image list, and then click Import Image.
3. In the Open dialog box, browse to and select the image to import. Click Open.
4. In the Apply To section, select Both.
5. Scroll to the bottom of the Image list and select the image you just imported. The image you selected should now be assigned to both the Small Image and Large Image fields of the command.
6. Click Apply to save the changes to the command.

As I mentioned earlier, the images used for your commands are stored in the CUIx file. You can manage the images stored in a CUIx file using the CUI Editor - Image Manager (see Figure 5.7). Click the Image Manager button in the Customizations In pane to display the CUI Editor - Image Manager. Here you can perform the following tasks:

- View the images and their sizes
- Import externally stored BMP files
- Export selected images in the CUIx file and save them as individual BMP files
- Remove the images that you are not currently using

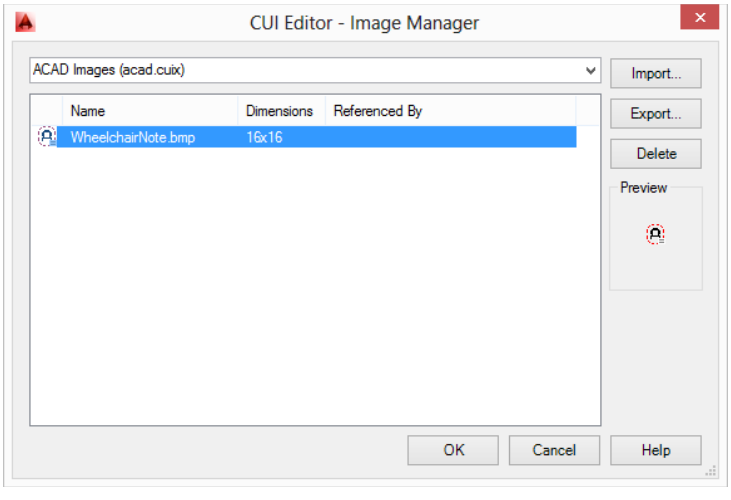


FIGURE 5.7 Managing images in the loaded CUIx files

Customizing User-Interface Elements

Out of the box, the AutoCAD user interface is designed for everyone, but not to accommodate the needs of any specific industry or any one single company's workflow. Many of the common user-interface elements in the AutoCAD application window can be customized, and you should take the time to customize them to get the most out of AutoCAD. You can add new elements that execute command macros and custom applications you create, remove or hide those that you do not use, or reorganize those that you use frequently to make them easier to access. You use the CUI Editor to modify the elements defined in a CUIx file.

The following elements of the user interface can be edited with the CUI Editor:

- Quick Access toolbar (QAT)
- Ribbon; panels and tabs
- Pull-down and shortcut menus
- Toolbars
- Double-click actions
- Shortcut and temporary override keys
- Mouse buttons
- Properties displayed as part of the Quick Properties palette or rollover tooltips
- Legacy elements; tablet menus and buttons; image tile and screen menus

Quick Access Toolbar

The Quick Access toolbar (QAT), shown in Figure 5.8, is part of the AutoCAD title bar area along the top of the application window. The tools commonly placed in the toolbar are for managing drawing files, plotting or publishing layouts, undoing or redoing recent actions, and setting a workspace current that is defined in the main CUIx file.

You can customize the QAT using one of the following methods:

QAT Customize Menu Clicking the Customize button on the far right side of the QAT displays the Customize menu. From this menu, you can toggle the display of several additional select commands, click More Commands to display the CUI Editor, or click Show Below/Above The Ribbon to change the placement of the QAT.

QAT Shortcut Menu Right-clicking a command or control on the QAT displays a shortcut menu that allows you to remove the element below the cursor, add a vertical separator bar to the right of the element under the cursor, click Customize Quick Access Toolbar to display the CUI Editor, or click Show Quick Access Toolbar Below/Above The Ribbon to change its placement.

Ribbon Button Right-clicking a command on the ribbon displays a shortcut that contains the Add To Quick Access Toolbar item. This item adds the command to the QAT.

CUI Editor The CUI Editor provides you with the same functionality that is found on the QAT's Customize and shortcut menus, and a few additional options as well. With the CUI Editor, you can choose to customize the default QAT or create a new one, and you can also change the order in which commands and controls are displayed.

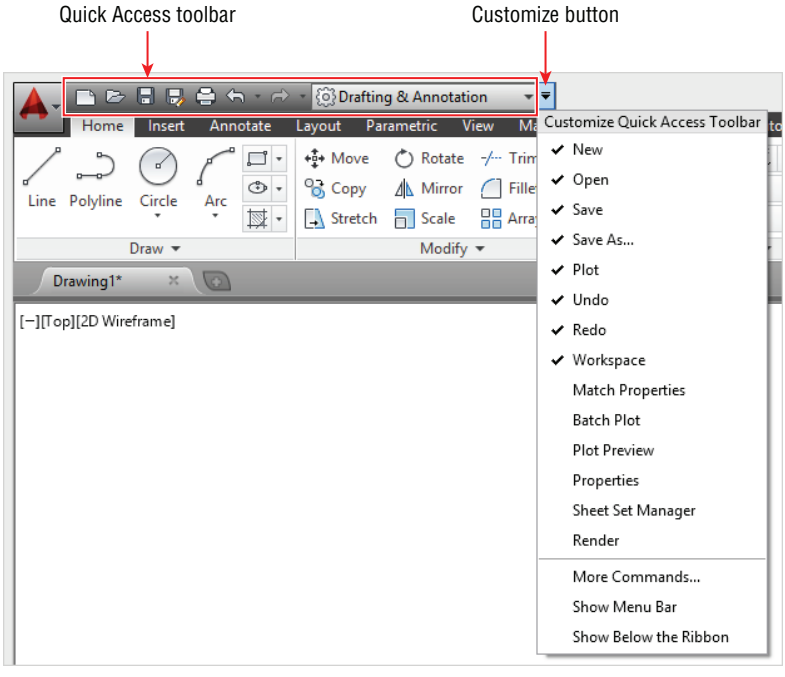


FIGURE 5.8 Accessing the customization options for the Quick Access toolbar

CUSTOMIZING THE DEFAULT QAT

The next example explains how to customize the default QAT. You will add the Wheelchair Note command that you created earlier in this chapter.

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane of the CUI Editor, expand the Quick Access Toolbars node. Expand the Quick Access Toolbar 1 node, or any other QAT you want to customize.
3. In the Command List pane, select Custom Commands from the Filter The Command List By Category drop-down list.
4. From the Command list, drag the Wheelchair Note command below the Ribbon Combo Box - Workspace control. Release the mouse button when the horizontal bar is displayed below the control (see Figure 5.9). After a command or control is added to the QAT, you can change how it is displayed by changing its properties under the Properties pane.

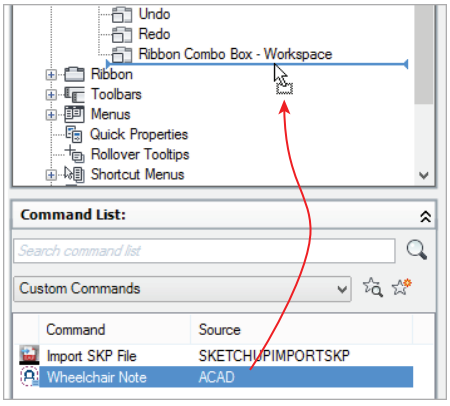


FIGURE 5.9
Adding a command to the QAT

5. Click Apply to save the changes.

Now that the Wheelchair Note command has been added, let's add a separator and Layer controls to the QAT that we are customizing in this exercise:

1. Use the command-list filter to access the Ribbon Control Elements. In the Command List pane, open the Filter The Command List By Category drop-down list and select Ribbon Control Elements (see Figure 5.10).

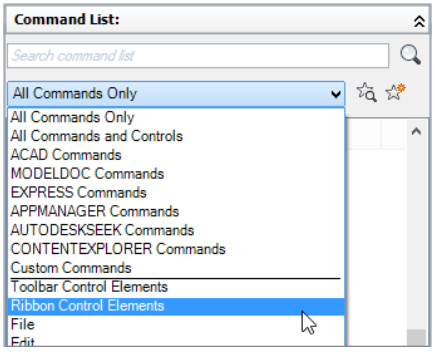


FIGURE 5.10
Accessing the controls that can be placed on the QAT

2. In the Search Command List text box (located just above the Filter The Command List By Category drop-down list), type **`layer`**.
3. In the command list, right-click Layer List Combo Box and select Copy.
4. In the Customizations In pane, right-click the Wheelchair Note command you added in Step 4. Click Paste. The Layer List Combo Box control is added to the QAT.
5. Right-click the Ribbon Combo Box - Workspace control under the QAT node and click Insert Separator. A separator element is added to the end of the toolbar.

6. Click Apply to save the changes.

There will be times when you want to remove access to particular commands. The next steps in this exercise explain how to remove the SaveAs command from the QAT and how to test the customized QAT:

1. Right-click the SaveAs element under the QAT and click Remove. In the message box, click Yes to remove the element.
2. Click and drag the Layer List Combo Box above the Wheelchair Note command in the tree view to reorder it on the QAT.
3. Click Apply to see the changes in the application window. The QAT and the elements under the QAT node in the CUI Editor should now look like Figure 5.11.

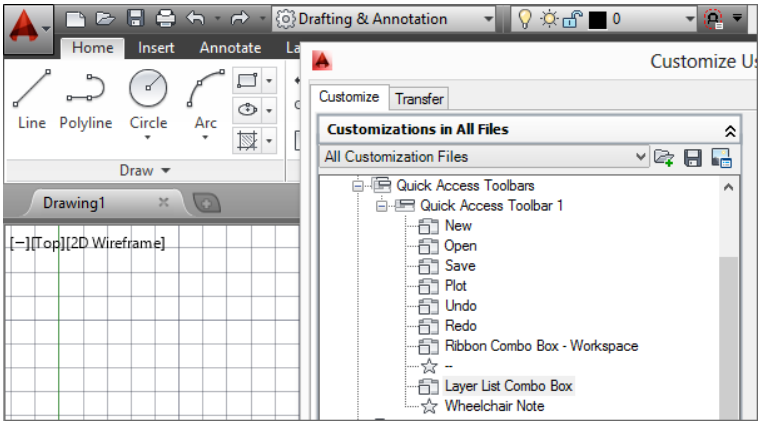


FIGURE 5.11 Results of the customization to the QAT

4. Click OK to save the changes to the CUIx file and return to the drawing window.
5. On the QAT, position the cursor over Wheelchair Note. Notice the contents of the tooltip, shown in Figure 5.12; you should see the information you entered when you created the command earlier in this exercise.

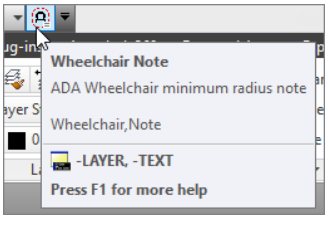


FIGURE 5.12 Tooltip for the custom Wheelchair Note command

6. Click Wheelchair Note and specify a point in the drawing window. The layer Notes is created and set as current, and two single-line text objects are created with the note (see Figure 5.13).

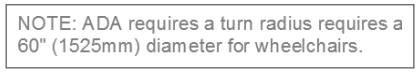


Figure 5.13
The results of using the Wheelchair Note command

NOTE You can create drop-down menus on the QAT that allow you to group multiple commands, not controls, into a single button. To create a drop-down menu, right-click the node of the QAT to which you want to add a drop-down menu in the CUI Editor and click New Drop-Down. Then, add commands to it from the Command List pane just as you did when you added commands to the QAT itself.

Creating a New QAT

While AutoCAD can display only a single QAT at a time, you could create your own QAT that contains the commands and controls you want to use instead of modifying the default toolbar that is defined in the `acad.cuix` file. Not only could you create your own new QAT, but you could define multiple QATs that contain different commands and controls for each department in your company or for each discipline of drawings that you work on. If you create a new QAT, you must assign it to a workspace to display it in the user interface.

These steps explain the overall process for creating a new QAT and displaying it within a workspace:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane of the CUI Editor, right-click the Quick Access Toolbars node and select New Quick Access Toolbar.
3. Enter a name for the new QAT or press Enter to accept the default name.
4. Customize the QAT as needed using the techniques introduced in the "Customizing the Default QAT" section.
5. Click OK to save the changes made. The new QAT must be added to a workspace before it can be displayed in the application window; see the "Organizing the User Interface with Workspaces" section later in this chapter for details on how to customize a workspace.

Ribbon

The main AutoCAD user-interface feature that you most likely have interacted with is the ribbon. The ribbon follows Microsoft's design concept called "fluent user interface" (or FUI) and is similar in concept to the one found in the Microsoft Office products. The idea behind the design is that it makes it easier to discover and access the commands and options that a user is looking for with a rich visual user experience.

Pull-down menus and toolbars, which most applications still use to this day, provide a tried-and-true user experience, but they work best with a somewhat limited selection of commands.

Pull-down menus and toolbars can handle hundreds of commands, but these elements lack the ability to start an operation and then give you additional choices while the operation is active.

Sure, you could show and hide menus and toolbars dynamically, but that introduces an element of inconsistency in the design. Where those items might appear this or next time becomes unpredictable. Dialog boxes are also a great way to allow the user to control the way a command or control might function, but that does not mean a user will always discover the most helpful settings. Instead of hiding useful settings and options, the design of the ribbon makes it easier to place them adjacent to a command or on a contextual tab. There is no doubt that the ribbon introduces an initial learning curve, but what doesn't when you are new to using it or after something you have done for a decade or two changes?

Commands and controls on the ribbon are organized by task through the use of tabs and panels, as shown in Figure 5.14. Each task is represented by a tab, which can hold different panels. A ribbon tab can be static—displayed all the time—or contextual, which means the tab is displayed only when a specific condition is met. If you have created a hatch or multiline text object in AutoCAD, chances are you worked with the Hatch Creation and Text Editor tabs that are displayed while you were using the `hatch` and `mtext` commands. Those tabs are contextual and are available only while the commands are active; ending the commands hides the tabs once again.

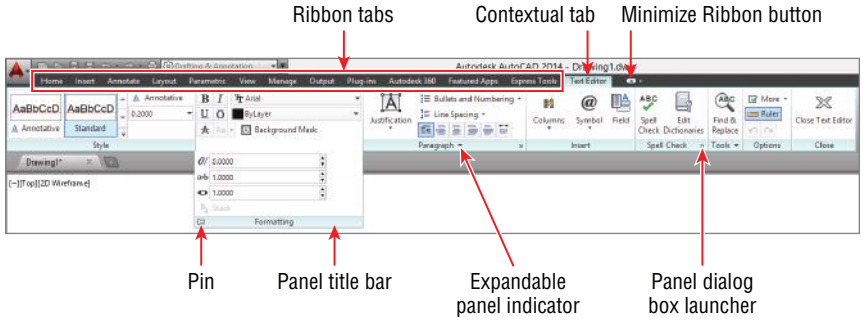


FIGURE 5.14 Command and control organization using the ribbon

The ribbon is divided into panels, which are used to organize and display commands and controls to the user. Panels have two different display states: normal and expanded. Not all panels are configured to be expanded, but when a panel offers additional commands or controls that are not displayed by default, you see a down arrow to the right of the ribbon's title. Clicking the panel's title bar expands the panel.

When a panel is expanded, it can be pinned (forced to remain expanded until you switch tabs or unpin it) using the Pin button, shown in Figure 5.14. Some panels show a panel dialog-box launcher button, also shown in Figure 5.14, which can start a command that commonly displays a dialog box or palette that is related to the commands and controls on the ribbon panel.

When customizing the ribbon, you can control the display of ribbon tabs with a workspace. Workspaces are also used to control the order in which tabs appear on the ribbon in the user interface. For more information on workspaces, see the section "Organizing the User Interface with Workspaces" later in this chapter. From the AutoCAD application window, you can also show and hide ribbon panels and tabs by right-clicking a panel or tab on the ribbon. Then, from the shortcut menu, click the element you want to show or hide. You can also modify the order in which ribbon panels and tabs are displayed by clicking and dragging an element on the ribbon.

Ribbon Panels

Ribbon panels are containers for the commands and controls that you eventually want to display on the ribbon. Each ribbon panel is divided into two areas; the upper area is always displayed, and the lower area is displayed only when the panel is expanded by clicking a panel title bar. The lower area is known as the *slideout*.

The commands and controls on a panel must be placed in a row. A panel can contain more than one row, but a panel must always have at least one row in order to contain commands or controls. A row can be divided into one or more rows with the use of a subpanel. A row can also contain a Fold panel, which can contain commands and controls as well. A fold panel differs from a subpanel in that it can't contain any rows; however, it can be assigned a collapse priority and resizing considerations. Separators and drop-down menus allow you to further organize related commands and controls on the panel. Figure 5.15 shows the Draw panel on the ribbon and how it is defined in the CUI Editor using rows and a single subpanel.

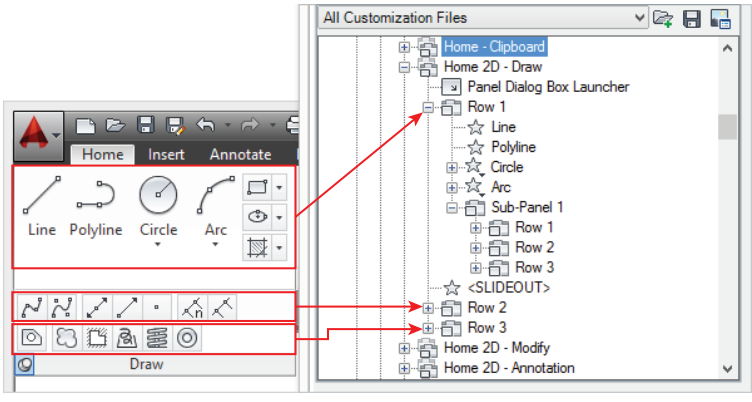


Figure 5.15
Structure of the Home 2D - Draw panel

The following example explains how to create a new panel named My Tools:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➤ Customization panel ➤ User Interface.
2. In the Customizations In pane of the CUI Editor, expand the Ribbon node.
3. Right-click the Panels node and select New Panel.
4. In the in-place text editor, type **`My Tools`** for the name of the panel and press Enter.
5. Click Apply to save the new ribbon panel.

Now, let's add structure to organize commands and controls:

1. Right-click the My Tools panel and click New Row. The new row is added below <SLIDEOUT>, which is the element that separates the rows that are displayed by default and those that are displayed when the panel is expanded.
2. Right-click Row 1 under the My Tools panel node and click New Sub-Panel.
3. Right-click Sub-Panel 1 under the Row 1 node and click New Row.

4. Under Sub-Panel 1, right-click the first row node and click New Drop-Down. Your ribbon panel should now look like Figure 5.16.

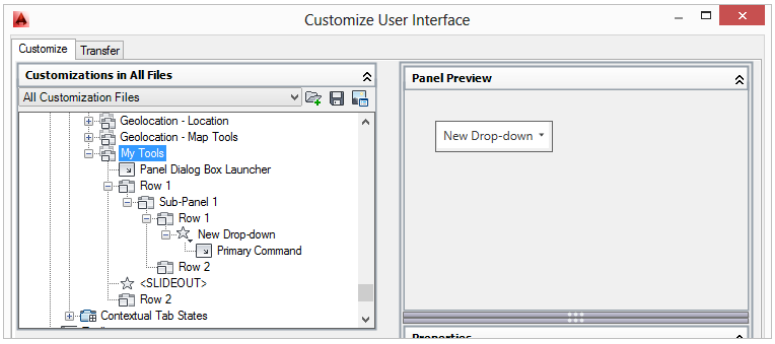


FIGURE 5.16 Structure for the new My Tools panel

5. Select the New Drop-Down node under Row 1 of Sub-Panel 1, and in the Properties pane, click the Button Style field. Select SmallWithText from the drop-down list that appears. Once a command or drop-down menu is added to a ribbon panel, you can customize how it will look using the properties available in the Properties pane.

The next steps explain how to add commands and controls to the rows of the ribbon panel:

1. In the Command List pane, click the Filter The Command List By Category drop-down list and select All Commands Only.
2. Locate the Wheelchair Note command and add it to Row 1 under the My Tools panel. (Type **`Wheelchair`** in the Search Command List text box to make it easier to locate the command.) Click and drag the Wheelchair Note command to the My Tools panel. When the cursor is over Row 1, release the mouse button. Then, drag the command so it is placed above Sub-Panel 1 and under Row 1.
3. Select the Wheelchair Note command under Row 1, and in the Properties pane, click the Button Style field. Select Large With Text (Vertical) from the drop-down list that appears. Once a command is added to a ribbon panel, you can customize how it will look using the new properties available in the Properties pane.
4. In the Command List pane, locate and add the Multileader, Multileader Edit, and Multileader Edit Remove commands to the New Drop-Down node under Row 1 of Sub-Panel 1. Remember, you can use the Search Command List text box to filter the Command List pane.

TIP You can press and hold the Ctrl key to select multiple commands. The order in which the commands are selected determines the order in which they are added to the panel.

5. Add the Multileader Style Manager command to the Panel Dialog Box Launcher node under the My Tools panel.

6. Add the other five multileader-related commands to Row 2 (located under the <SLIDEOUT> item).
7. Click the Filter The Command List By Category drop-down list and select Ribbon Control Elements. Add the Ribbon Combo Box - Multileader Style control to Row 2 under Sub-Panel 1.
8. Add some of your favorite commands and controls that you use often to the ribbon panel, if you want.
9. Click Apply to save the new panel. The new panel should appear in the CUI Editor, as shown in Figure 5.17.

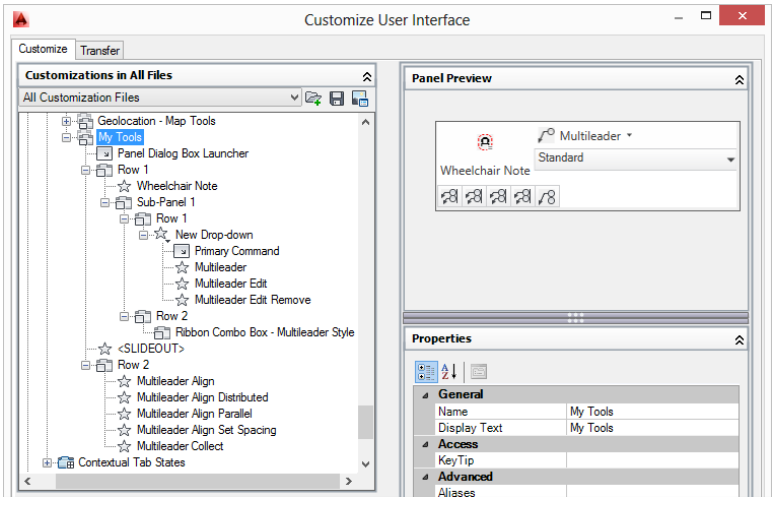


FIGURE 5.17
The completed My Tools panel

Once you have created a panel, you must add it to a tab in order for it to be displayed on the ribbon. I cover customizing ribbon tabs in the next section. You can modify an existing panel by selecting it from the Panels node under the Ribbons node of the Customizations In pane. The process for modifying a panel is similar to the process you used when you created the My Tools panel.

Ribbon Tabs

Ribbon tabs are used to control the display and organization of ribbon panels in the user interface. You can add and remove ribbon panels to or from one of the standard ribbon tabs or create your own. Often, if you create your own panels you will want to create your own ribbon tabs as well. Several conditions determine whether a tab displays in the user interface:

- Is the tab part of the current workspace?
- Is the tab enabled?

- Has the tab been assigned to a contextual state?
- Has a contextual condition been met?

Creating a New Ribbon Tab

The following example explains how to create a new tab named Favorites and how to add several ribbon panels to the new tab:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane of the CUI Editor, expand the Ribbon node.
3. Right-click the Tabs node and select New Tab.
4. Using the in-place text editor, type **`My Favorites Tab`** for the name of the new tab and press Enter.
5. Select the My Favorites Tab from the Ribbon ➢ Tabs node, and in the Properties pane, change the value in the Display Name field to **`Favorites`**. The Display Name field controls the text that appears in the user interface as the tab label.
6. In the Customizations In pane, go to the Ribbon node and expand Panels. Select the My Tools node and right-click. Click Copy.
7. Under the Tabs node, select the Favorites node and right-click. Click Paste. A reference to the My Tools panel is added to the Favorites tab. Figure 5.18 shows what the Favorites tab should look like in the Customizations In pane, and the tab's settings in the Properties pane.

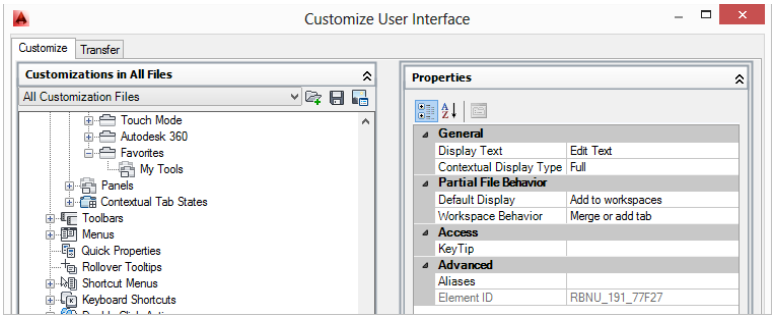


FIGURE 5.18 Favorites tab with the My Tools panel

8. Click Apply to save the new tab. The new tab is not added to the ribbon until it has been added to a workspace or a contextual tab state.

Displaying Ribbon Tabs

After a ribbon tab has been created, you have two options for displaying it in the user interface:

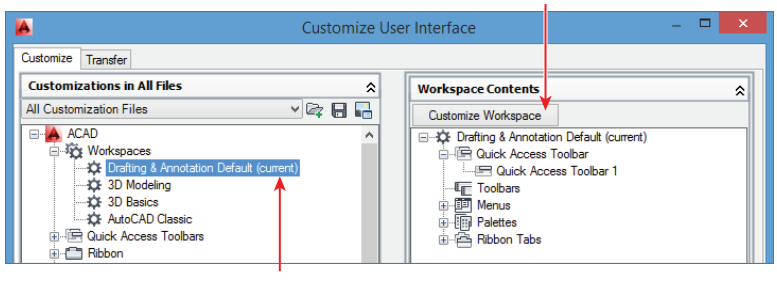
- Add the ribbon tab to a workspace
- Add the ribbon tab to a contextual tab state

When you choose to add it to a workspace, you can control the location in which the tab appears on the ribbon and its default display state: shown or hidden. If you add the tab to a contextual tab state, you can control the tab's display type: full or merged.

Use the following steps to add the Favorites tab to the ribbon for the current workspace:

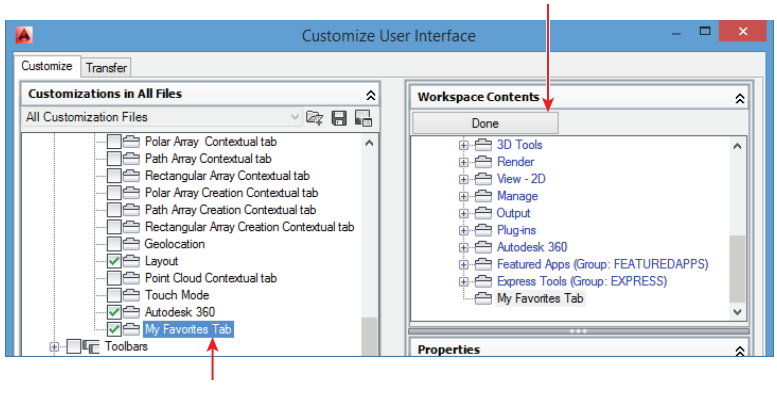
1. In the Customizations In pane, expand the Workspaces node and select the workspace that ends with the text `(current)`.
2. In the Workspace Controls pane, click Customize Workspace, as shown in Figure 5.19.

FIGURE 5.19
Customizing the current workspace



3. In the Customizations In pane, expand Ribbon ➤ Tabs and click My Favorites Tab (shown in Figure 5.20). The My Favorites Tab check box should be selected.

FIGURE 5.20
Adding the ribbon tab to the current workspace



4. In the Workspace Contents pane, click Done.
5. Expand the Ribbon Tabs node and drag Favorites above Home - 2D (shown in Figure 5.21).
6. Click OK to save the changes and close the CUI Editor.
7. On the ribbon, click the Favorites tab (shown in Figure 5.22). Test the various parts of the ribbon panel; click the panel's title bar to expand the panel and display the dialog box launcher button.

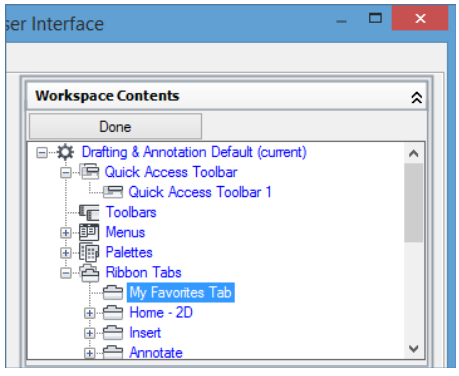


FIGURE 5.21 Controlling the order of the ribbon tab with the workspace

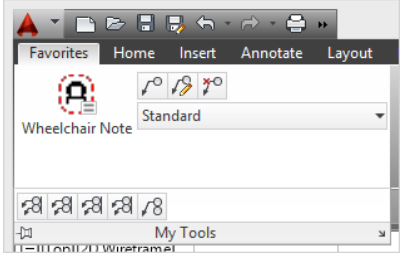


FIGURE 5.22 The Favorites tab displayed on the ribbon

Any ribbon tab can be added to a contextual tab, but typically the scope of the commands and controls on the panels of the tab are limited to editing objects only. Contextual tab states are not workspace specific, so there is no need to add contextual tabs to a current workspace.

The following example explains how to create a new ribbon tab that contains the Annotate - Text panel, and adds the new ribbon tab to the Text, Multiline Selected contextual tab state.

1. In the drawing window, use the `mtext` command to create a multiline text object. Now, select the text object. Notice that no contextual tab is displayed unless the multiline text object is being edited in the in-place text editor.
2. In the CUI Editor, create a new ribbon tab named **`Text Contextual tab`**.
3. Select the Text Contextual tab from the Ribbon ➢ Tabs node, and then in the Properties pane, change the value of the Display Text field to **`Edit Text`**.
4. Click the Contextual Display type field, and select Full from the drop-down list. Full designates that the panels associated with the tab are displayed on their own tab, while Merged designates that the panels associated with the tab are displayed no matter which ribbon tab is current.
5. In the Customizations In pane, go to the Ribbon node and expand Panels. Select and then right-click the Annotate - Text panel. Click Copy.
6. Under the Ribbon ➢ Tabs node, select the Text Contextual tab and right-click. Click Paste.
7. In the Customizations In pane, expand Ribbon ➢ Contextual Tab States.

8. From the Ribbon ➢ Tabs node, drag the Text Contextual tab to the Text, Multiline Selected contextual tab state.
9. Click OK to save the changes and close the CUI Editor.
10. In the drawing window, select the multiline text object. The Edit Text tab is displayed with the Text panel, as shown in Figure 5.23. By default, tabs that are assigned to contextual tab states are added to the right side of the ribbon, but in Figure 5.23 the Edit Text tab was dragged to the left side directly in the application window.

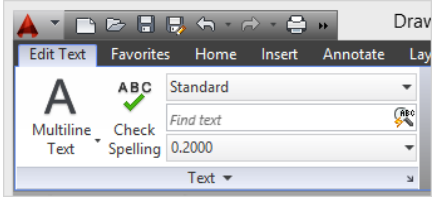


FIGURE 5.23 The custom Edit Text tab displayed when the multiline text object was selected

Pull-Down Menus

The menu bar is an area commonly found along the top of an application window. It contains a number of menus, which are also referred to as *pull-down menus*. You click a label on the menu bar to display the associated pull-down menu. When displayed, you can click one of the items on the pull-down menu to start a command. A pull-down menu can contain separators and submenus to group related commands together.

The menu bar and pull-down menus were the primary method for accessing commands before the introduction of the ribbon. They are still used for corporate customization and are displayed by default with the AutoCAD Classic workspace. It is possible to display both the ribbon and menu bar. When the `menubar` system variable is set to 1, the menu bar is displayed just below the title bar of the AutoCAD application window.

Like the QAT and ribbon, you can customize the pull-down menus that come with AutoCAD or create your own. Pull-down menus can contain the commands that can be added to other user-interface elements but can't contain a drop-down list, check box, text box, or other common Windows controls that can be found on the ribbon or some toolbars. You can, however, use DIESEL to enable, disable, and/or display a check mark next to a menu item. For information on DIESEL and how to use it, refer to the AutoCAD Help system.

As you have seen with other user-interface elements, workspaces are also used to control the display of a pull-down menu on the menu bar and in which order all pull-down menus should be displayed. For more information on workspaces, see the "Organizing the User Interface with Workspaces" section later in this chapter.

The following explains how to create a new pull-down menu, add commands, and organize the commands with a separator and submenu:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane of the CUI Editor, right-click the Menus node and select New Menu.

3. In the in-place text editor, type **`Favorites`** for the name of the new pull-down menu and press Enter.
4. In the Command List pane, locate the Wheelchair Note command. (Type **`wheelchair`** in the Search Command List text box.) Click the command name and then drag the Wheelchair Note command to the Favorites pull-down menu under the Menus node in the Customizations In pane. When the cursor is over Favorites, release the mouse button.
5. In the Customizations In pane, under the Menus node, right-click the Favorites pull-down menu and click New Sub-menu. In the in-place text editor, type **`Annotation Tools`** for the name of the new submenu and press Enter.
6. In the Command List pane, locate the Multiline Text and Single Line Text commands and drag them to the Annotation Tools submenu. (If you have difficulty finding the commands, type **`line text`** in the Search Command List text box.)
7. In the Customizations In pane, under the Menus node, right-click the Annotation Tools submenu under the Favorites pull-down menu and click Insert Separator.
8. In the Command List pane, locate the Multileader command and drag it below the separator on the Annotation Tools submenu. (If you have difficulty finding the command, type **`multileader`** in the Search Command List text box.)
9. Click Apply to save the new pull-down menu. Figure 5.24 shows what the completed pull-down menu should look like under the Menus node.

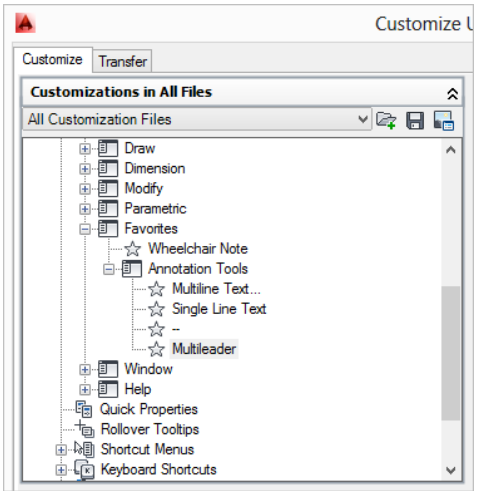


FIGURE 5.24 Structure of the Favorites pull-down menu in the CUI Editor

While a new pull-down menu is added to all the workspaces currently defined in the main customization (CUIx) file, you will want to make changes to each workspace to ensure the pull-down menu is in the correct position on the menu bar.

Use the following steps to add the pull-down menu if it is not displayed in the current workspace. I will also show you how to change the menu's position.

1. In the Customizations In pane, expand the Workspaces node and select the workspace that ends with the text `(current)`.

2. In the Properties pane, click the Menu Bar field and select On from the drop-down list. Enabling this property ensures that the menu bar is displayed when the workspace is set as current; this sets the `menubar` system variable to a value of 1.
3. In the Workspace Contents pane, click Customize Workspace.
4. In the Customizations In pane, expand Menus and click Favorites if it is not already checked.
5. In the Workspace Contents pane, expand the Menus node and drag Favorites above Window. Click Done.
6. Click OK to save the changes and close the CUI Editor.
7. On the menu bar, click Favorites. Test the various menu items and the subpanel pull-down menu. Figure 5.25 shows what the Favorites pull-down menu looks like on the menu bar.

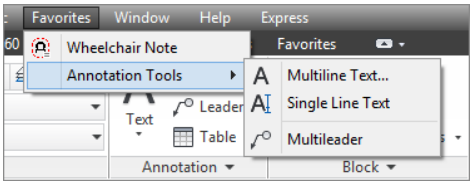


FIGURE 5.25 Favorites pull-down menu on the menu bar

Shortcut Menus

Shortcut menus make it easy to access the commands that you need when you need them—and near the cursor, so you don't have to leave the drawing area. You display a shortcut menu by right-clicking or secondary-clicking on the input device. The commands that you can access from a shortcut menu are often determined by the context in which the menu is displayed, which is why shortcut menus are sometimes referred to as *context or contextual menus*. Table 5.3 lists the shortcut menus that you can customize and the contexts in which they are displayed.

TABLE 5.3: Customizable shortcut menus

CONTEXT	DESCRIPTION
Hot Grip	Displayed when a grip has been selected and is ready for editing, and you right-click in the drawing area
Object Snap	Displayed when the Shift key is held and you right-click in the drawing area
Default Mode	Displayed when no command is active, an object is selected, grip editing is not active, and you right-click in the drawing area
Command Mode	Displayed when a command is active and you right-click in the drawing area
Edit Mode	Displayed when an object is selected and you right-click in the drawing area

NOTE The `shortcutmenu` system variable controls the display of the shortcut menus related to the Default, Command, and Edit modes. These settings can also be controlled from the Windows Standard Behavior section of the User Preferences tab in the Options dialog box (`options` command). In addition to controlling which menus are displayed, you can control the duration for which the right pointer device button needs to be held before the shortcut menu is displayed. You can change the duration by using the Options dialog box or the `shortcutmenuduration` system variable.

Shortcut menus are customized with the CUI Editor using techniques that are nearly identical to those used for customizing pull-down menus. You add commands to the shortcut menu and use separators and submenus to organize related commands. Shortcut menus are not displayed as part of the main user interface, but are called on based on the current context. AutoCAD uses a special property value called an alias to determine which shortcut menu should be displayed. Each alias must be unique inside a customization (CUIx) file.

Table 5.4 lists the unique aliases and alias naming conventions that AutoCAD uses for displaying specific shortcut menus.

TABLE 5.4: Aliases and alias naming conventions for shortcut menus

ALIAS	DESCRIPTION
`GRIPS`	Hot Grip Cursor menu
`SNAP,POP0`	Object Snap Cursor menu
`CMDEFAULT`	Default Mode menu
`CMCOMMAND`	Command Mode menu
`CMEDIT`	Edit Mode menu
`COMMAND_`*cmdname*	Command-specific menu; *cmdname* represents the name of the command that the shortcut menu should be associated with.
`OBJECT_objectname` or `OBJECTS_`*objectname*	Single or multiple selected objects menu; *objectname* represents the type of the object that the shortcut menu should be associated with.

The next example explains how to create a new object shortcut menu that adds items to the Edit Mode menu when a line is selected:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane in the CUI Editor, right-click the Shortcut Menus node and select New Shortcut Menu.
3. In the in-place text editor, type **`Line Objects Menu`** for the name of the new shortcut menu and press Enter.

4. Select the Line Objects Menu from the Shortcuts Menus node, and then go to the Properties pane and click the Aliases field. Click the ellipsis […] button to display the Aliases dialog box. Click after the alias in the text box and press Enter. Type **OBJECTS_LINE** and click OK. For information on defining command- and object-related shortcut menus, see the "Command Mode Shortcut Menus" and "Object Mode Shortcut Menus" sections later in this chapter.
5. In the Command List pane, locate the Stretch, Trim, and Extend commands and drag them to the Line Objects Menu item in the Shortcut Menus node.
6. Click OK to save the new shortcut menu.
7. In the drawing window, create a few lines.
8. Select the lines you create and right-click. The shortcut menu with the CMEDIT alias is displayed and the items in the Line Objects Menu shortcut menu are merged with it, as you can see in Figure 5.26.

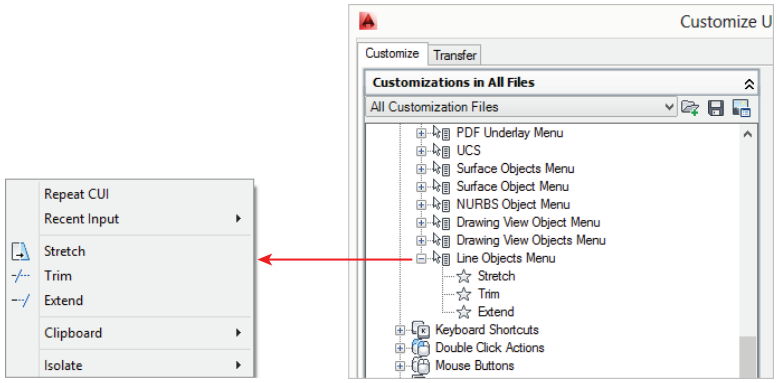


FIGURE 5.26
Line Objects Menu in the drawing window

Command Mode Shortcut Menus

Command-specific menus insert additional items into the shortcut menu with the CMCOMMAND alias. You use the COMMAND_*cmdname* alias to specify which command the shortcut menu should be associated with. *cmdname* must match the name of a command defined with ObjectARX® or Managed .NET, not one that has been defined with AutoLISP. For example, to associate items with the LINE command's shortcut menu you would create a shortcut menu with the alias COMMAND_LINE.

Object Mode Shortcut Menus

Single or multiple selected object menus insert additional items into the shortcut menu with the CMEDIT alias. You use the OBJECT_*objectname* and OBJECTS_*objectname* aliases to specify which object type the shortcut menu should be associated with. OBJECT_*objectname* is used when you select a single object of a specific object type and right-click in the drawing area; OBJECTS_*objectname* is used when multiple objects of a single object type are selected.

For example, to associate items with the Edit Mode shortcut menu when a single arc object is selected, you would create a shortcut menu with the alias OBJECT_ARC.

If a shortcut menu with the alias `OBJECT_`*objectname* is not defined but a shortcut menu with the alias `OBJECTS_`*objectname* is, `OBJECTS_`*objectname* applies to the context of when one or more objects of the specified object type are selected. *objectname* must match a valid DXF Code 0 value. You can search the product Help system on the keywords *DXF Entities* to locate a listing of values for standard AutoCAD objects.

There are some additional names, though, that do not match a DXF Code 0 value. These exceptions apply to block references that have the value `INSERT`. Table 5.5 lists the additional names that are used to identify types of block reference objects.

TABLE 5.5: Special names used to identify types of block reference objects

NAME	DESCRIPTION
`ATTBLOCKREF`	Block reference containing attribute references
`ATTDYNBLOCKREF`	Dynamic block reference containing attribute references
`BLOCKREF`	Block reference without attribute references
`DYNBLOCKREF`	Dynamic block reference without attribute references
`XREF`	External drawing reference (xref)

TIP If you are unsure what the DXF Code 0 value is for an object, place the object in the drawing window and enter the AutoLISP expression **`(cdr (assoc 0 (entget (car (entsel)))))`** at the AutoCAD command prompt. When prompted, select the object and the DXF name and the entity will be displayed.

Toolbars

Toolbars can be found in most Windows applications. Before the ribbon or even the dashboard, which was the predecessor to the ribbon, toolbars were one of the earliest visual user-interface elements. Buttons on a toolbar can be clicked to start a command or pressed and held to display a flyout. A flyout on a toolbar is similar to a submenu on a pull-down menu or even a drop-down menu on the ribbon, but flyouts are based on other toolbars, as defined in a loaded customization (CUIx) file. Toolbars can also contain controls, such as drop-down lists or text boxes, and separators to organize commands and controls.

While toolbars for the most part are no longer one of the primary user-interface elements in AutoCAD, they are still useful for corporate customization and are displayed by default with the AutoCAD Classic workspace. It is possible to display toolbars and the ribbon at the same time; this can be helpful if you want access to specific commands and controls no matter which ribbon tab is active. For example, you might consider displaying the Layers toolbar so you can change the current layer without switching to the Home tab on the ribbon.

Workspaces are used to control the display of toolbars, as well as their position in the user interface. For more information on workspaces, see the "Organizing the User Interface with Workspaces" section later in this chapter.

The following example explains how to create a new toolbar and add an existing toolbar as a flyout:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➤ Customization panel ➤ User Interface.
2. In the Customizations In pane of the CUI Editor, right-click the Toolbars node and select New Toolbar.
3. In the in-place text editor, type **`Favorites`** for the name of the new toolbar and press Enter.
4. In the Command List pane, locate and add the Wheelchair Note command to the Favorites toolbar. (Type **`wheelchair`** in the Search Command List text box to make it easier to find the command.) Click and drag the Wheelchair Note command to the Favorites toolbar under the Toolbars node in the Customizations In pane. When the cursor is over Favorites, release the mouse button.
5. Locate and add the Multileader command below the Wheelchair command. (Type **`multileader`** in the Search Command List text box to make it easier to find the command.)
6. In the Customizations In pane, under the Toolbars node click and drag the Text toolbar between the Wheelchair and Multileader commands in the Favorites toolbar. This creates a flyout on the Favorites toolbar containing the commands of the Text toolbar. A toolbar can contain controls, but when used as a flyout, the controls are not displayed.
7. Click Apply to save the new toolbar. Figure 5.27 shows what the completed toolbar should look like under the Toolbars node.

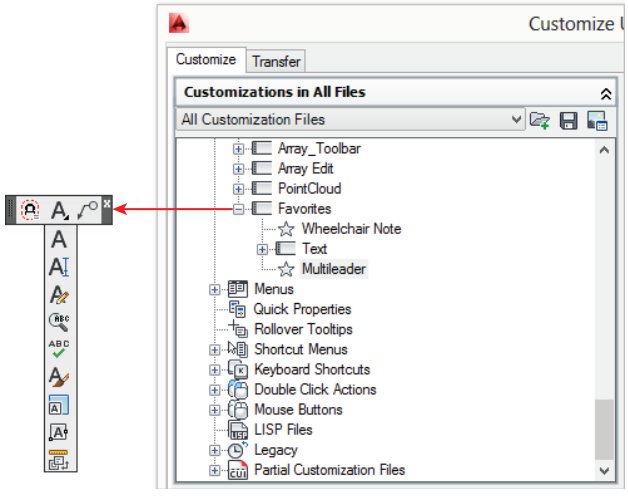


FIGURE 5.27 Structure of the Favorites toolbar in the CUI Editor and how it appears in the user interface

When a toolbar is created, it is added to all the workspaces currently defined in the customization (CUIx) file. Use the following steps to add the Favorites and Layers toolbars to the current workspace:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane, expand the Workspaces node and select the workspace that ends with the text `(current)`.
3. In the Workspace Contents pane, click Customize Workspace.
4. In the Customizations In pane, expand Toolbars and click Favorites if it is not already checked. Click Layers as well if it is not already checked.
5. In the Workspace Controls pane, expand the Toolbars node and select Favorites.
6. In the Properties pane, you can edit the Orientation, Location, and Rows properties of the toolbar. Click Done.

TIP Rather than controlling the orientation and location of toolbars using the CUI Editor, you can drag and position toolbars in the application window. Once the toolbars are positioned, use the `wssave` command to save the changes to a workspace.

7. Click OK to save the changes and close the CUI Editor.
8. Click the Wheelchair Note and Multileader to start the command macros. Click and hold the mouse button over the Text button in the middle of the toolbar that represents the Text toolbar you added in the previous exercise. Then drag the cursor on the flyout and release the mouse button when it is over the button of the macro to start.

Shortcut and Temporary Override Keys

Shortcut and temporary override key combinations are used to execute a command macro. A temporary override key combination can also execute a second macro when the key combination is released. Both key types require you to define a key combination that includes at least the Shift, Ctrl, or Alt key and, in almost all combinations, one or more of the standard and virtual keys on the keyboard.

Discovering Existing Shortcut and Temporary Override Keys

The CUI Editor allows you to print or copy a list of all the shortcut and temporary override keys to your default printer or the Windows Clipboard. Both of these operations can be helpful to let users know which key combinations are available to them and can be performed by doing the following:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane of the CUI Editor, select the Keyboard Shortcuts node.

3. In the Shortcut Keys pane, click the Type drop-down list and choose the type of keys you want to list:
   - All keys
   - Accelerator (shortcut) keys
   - Temporary Override Keys
4. Click the Status drop-down list and choose the status of the keys you want to list:
   - All
   - Active
   - Inactive
   - Unassigned
5. Do one of the following:
   - Click Copy To Clipboard to copy a tab-delimited list of all the keys currently displayed in the list.
   - Click Print to output a list of all the keys currently displayed in the list.
6. Click OK to exit the CUI Editor.

Not all key combinations are included in the list. Those that are common Windows shortcut keys are not defined as part of a customization (CUIx) file. You can search the product's Help on the keywords *shortcut keys reference* and *temporary keys reference* to locate listings of both keyboard-shortcut types.

The following example explains how to create a shortcut key that starts the Wheelchair Note command created earlier in this chapter:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane of the CUI Editor, expand the Keyboard Shortcuts node.
3. Go to the Command List pane, locate the Wheelchair Note command, and drag it to the Shortcut Keys node in the Customizations In pane. (If you have difficulty locating the command, type **`wheelchair`** in the Search Command List text box.)
4. With the Wheelchair Note command highlighted under the Shortcut Keys node in the Properties pane, and click in the Key(s) field. Click the ellipsis […] button to display the Shortcut Keys dialog box. Click in the Press The New Shortcut Key text box, and then press and hold the Ctrl, Shift, and N keys. CTRL+SHIFT+N should now appear in the text box. Click OK.
5. Click OK to save the changes and close the CUI Editor.
6. In the drawing window, press the key combination Ctrl+Shift+N.
7. When prompted, specify a point in the drawing.

You can use these steps to create a new temporary override key that toggles the current setting of the osnapz system variable:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane, right-click the Keyboard Shortcuts node and click New Temporary Override.
3. In the in-place text editor, type **Object Snap Z Toggle** as the name of the new temporary override and press Enter.
4. Select the Object Snap Z Toggle temporary override under the Temporary Override Keys node in the Properties pane, and click in the Key(s) field. Click the ellipsis […] button to display the Shortcut Keys dialog box. Click in the Press The New Shortcut Key text box, and then press and hold the Shift and F keys. SHIFT+F should now appear in the text box. Ctrl and Alt cannot be used when defining the key combination for a temporary override key. Click OK.
5. Click in the Macro 1 (Key Down) field and replace the default text by typing the following: **^P'_.osnapz $M=$(if,$(and,$(getvar,osnapz),1),0,1)**. The macro toggles the current value of the osnapz system variable when the key combination is held and then changes it again when the key combination is released.
6. Click OK to save the changes and close the CUI Editor.
7. In the drawing window, draw a 3D box with the box command.
8. Enable the Endpoint running object snap.
9. Start the line command and position the crosshairs close to the top corner of the 3D box. The Endpoint marker should appear. Click, and the line should start from that endpoint. Cancel the line command.
10. Start the line command again. This time press and hold the key combination Shift+F and position the crosshairs close to the top corner of the 3D box. The Endpoint object snap marker should appear on the work plane. If the toggle does not seem to work correctly, you might need to disable Dynamic UCS in order for the temporary override key to work properly.

TIP The tempoverride system variable needs to be set to a value of 1 in order to use temporary override keys.

Double-Click Actions

Double-click actions, as the name implies, are actions performed when you double-click something; in this case, that something happens to be a drawing object in the drawing window. While a double-click action starts a command macro that is defined in the Command List pane, all of the commands assigned to the default double-click actions edit the drawing object that was double-clicked.

When possible, a specific object-related editing command is started. For example, `mtedit` is started when you double-click a multiline text object, or `pedit` is started for a polyline object. If an object does not have a double-click action defined in the main customization (CUIx) file, the Properties palette is displayed. As a double-click action is defined, you must specify the type of object that the double-click action should be performed on using the object's DXF Code 0 value. You can figure out the DXF 0 Code value for an object by using the information I mentioned earlier, in the "Object Mode Shortcut Menus" section.

The following explains how to create a double-click action for an RTEXT object that is created with the `rtext` command that is part of Express Tools:

1. On the ribbon, click Express Tools tab ➢ Text panel, click the panel's title bar, and then click Remote Text. You can also enter **`rtext`** at the command prompt and press Enter.
2. At the `Enter an option [Style/Height/Rotation/File/Diesel] <Diesel>:` prompt, enter **`d`** and press Enter.
3. In the Edit RText dialog box, type **`Filename: $(getvar,dwgname)`** and click OK.
4. At the `Specify start point of RText:` prompt, specify a point in the drawing window.
5. At the `Enter an option [Style/Height/Rotation/Edit]:` prompt, press Enter to exit the command.
6. Double-click the new remote text object; you should see the Properties palette displayed even though there is a command named `rtedit` that allows you to edit remote text.
7. At the command prompt, enter **`(cdr (assoc 0 (entget (car (entsel)))))`** and press Enter. Select the remote text object, and the text `RTEXT` is returned. RTEXT is the object name of a remote text object; this will be needed to create the double-click action.
8. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
9. In the CUI Editor, from the Command List pane select Create A New Command.
10. In the Properties pane, type **`Remote Text Edit`** in the Name field.
11. In the Macro field, type **`._rtedit;_e;`**. If you opened the Long String Editor, click OK.
12. In the Customizations In pane, right-click the Double Click Actions node and click New Double Click Action.
13. In the in-place text editor, type **`Rtext`** for the name of the new double-click action and press Enter.
14. Select the Rtext item under the Double Click Actions node; in the Properties pane, click in the Object Name field and type **`RTEXT`**.
15. In the Command List pane, locate the Remote Text Edit command and then add it to the new Rtext item under the Double Click Actions node. (Type **`remote text edit`** in the Search Command List text box.) Click and drag the Remote Text Edit command to the

Rtext item under the Double Click Actions node. When the cursor is over the Rtext item, release the mouse button.

16. Click OK to save the changes and close the CUI Editor.
17. Double-click the remote text object that you created in the first five steps. The Edit RText dialog box is displayed with the DIESEL expression that was added to the remote text object.

Other Elements

The AutoCAD user experience has changed and evolved over the years, and so have the elements of the user interface. The ribbon and Quick Access toolbar (QAT) for the most part have replaced the use of pull-down menus and toolbars that were the primary ways of accessing commands for over a decade. The user-interface elements that I have covered in this chapter are the main user-interface elements that are most frequently used and customized. There are a few others that are not as frequently customized or are basically retired from the product. These other user-interface elements include the following:

Mouse Buttons You can specify the actions assigned to each button on your pointing device, with the exception of the left or primary mouse button, which is always the pick button. In addition to customizing the basic click event performed by a mouse button, you can customize the click action taken when the Shift and Ctrl keys are held while pressing a mouse button.

Tablet Buttons and Menus You can specify the actions assigned to each button on your tablet's pointer device, with the exception of the primary mouse button, which is always the pick. Some tablet pointer devices support up to 16 buttons. In addition to customizing the basic click event performed by a mouse button, you can customize the click action taken when the Shift and Ctrl keys are held while pressing a mouse button.

Tablet menus allow you to map which area of the tablet represents the digitizing and drawing areas, in addition to where you can click to start a command. To help users identify each of the areas and commands, you create an overlay that sits on top of the tablet.

Image Tile Menus Image tile menus (also known as *icon tile menus*) allow you to associate a slide image with a command macro. The image tile menu was one of the first user interfaces that allowed you to associate an image with a command macro. The images used for the menus were created with the `mslide` command. Multiple slide images can be combined into a slide library and used with the image tile menu. It is common to see image tile menus used to insert blocks, but they can be used to pass values to an AutoLISP or other custom program.

Screen Menus The screen menu is a user interface that is kind of like a stack of papers. Clicking an item on the screen menu can execute a command macro or jump to a different page in the screen menu. Until recently, the screen menu was one of the oldest active user-interface elements that you could use and customize, but it has recently been retired. The screen menu can be displayed and customized after the `screenmenu` system variable has been redefined using the `redefine` command in AutoCAD 2014. Earlier releases do not require you to redefine the `screenmenu` system variable.

Dashboard Panels The Dashboard palette was removed from the product in AutoCAD 2009. If you are migrating from an earlier release, you can convert a dashboard panel to a

ribbon panel. Dashboard panels, if they exist in a customization file, can be converted on the Transfer tab of the CUI Editor.

Refer to the product Help system for additional information on how to customize these user-interface elements or migrate them from a previous release so they can be used in the latest release.

Setting Up Rollover Tooltips and the Quick Properties Palette

Rollover tooltips and the Quick Properties palette, shown in Figure 5.28, allow you to quickly query and change the property values of an object. Instead of having to select an object to view its properties, you can position the cursor over an object to see the current value of several properties for an object in a tooltip. Which properties are displayed for the rollover tooltip can be customized using the CUI Editor.

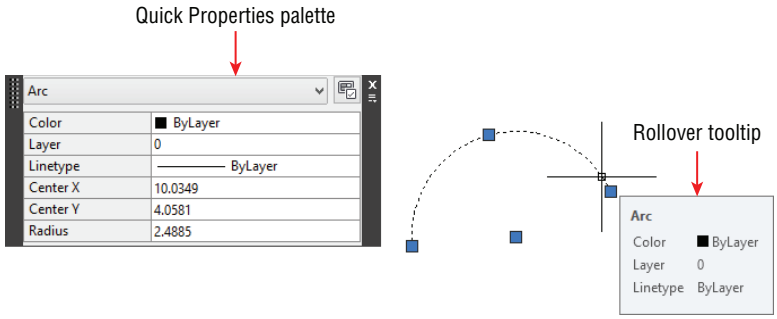


FIGURE 5.28 Rollover tooltip and Quick Properties panel displaying the properties for an arc

The Quick Properties palette (`quickproperties` command) is also a convenient way to query the property values of an object, but also to be able to edit values. You can specify which properties are displayed on the Quick Properties palette by using the CUI Editor, and you can control the appearance and behavior of the palette using the Quick Properties tab (see Figure 5.29) of the Drafting Settings dialog box (`dsettings` command).

NOTE Some of the settings on the Quick Properties tab of the Drafting Settings dialog box can also be changed using the `qpmode` and `qplocation` system variables.

Use the following steps to customize the properties displayed on the rollover tooltip (or Quick Properties palette) for a Hatch object:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➤ Customization panel ➤ User Interface.
2. In the Customizations In pane of the CUI Editor, select the Rollover Tooltips (or Quick Properties) node.
3. In the Dynamic pane, select Hatch from the Object Type list (see Figure 5.30). If Hatch is not displayed, click Edit Object Type List at the top of the Object Type list. In the Edit Object Type List dialog box, click Hatch and then click OK.

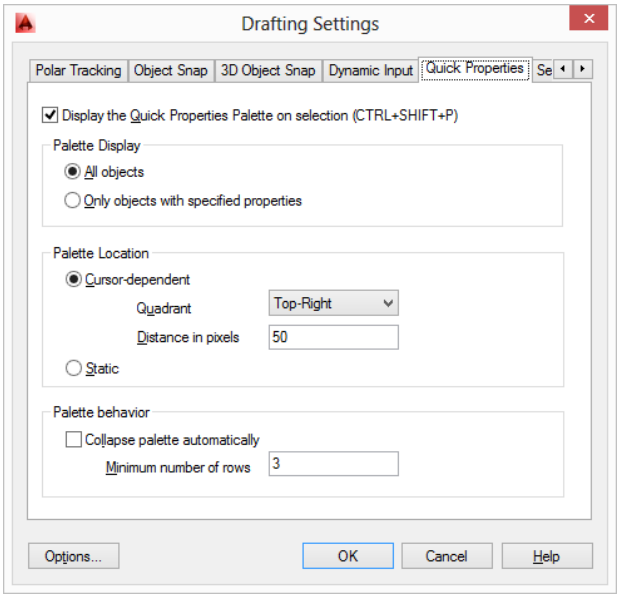


FIGURE 5.29
Controlling the appearance and behavior of the Quick Properties palette

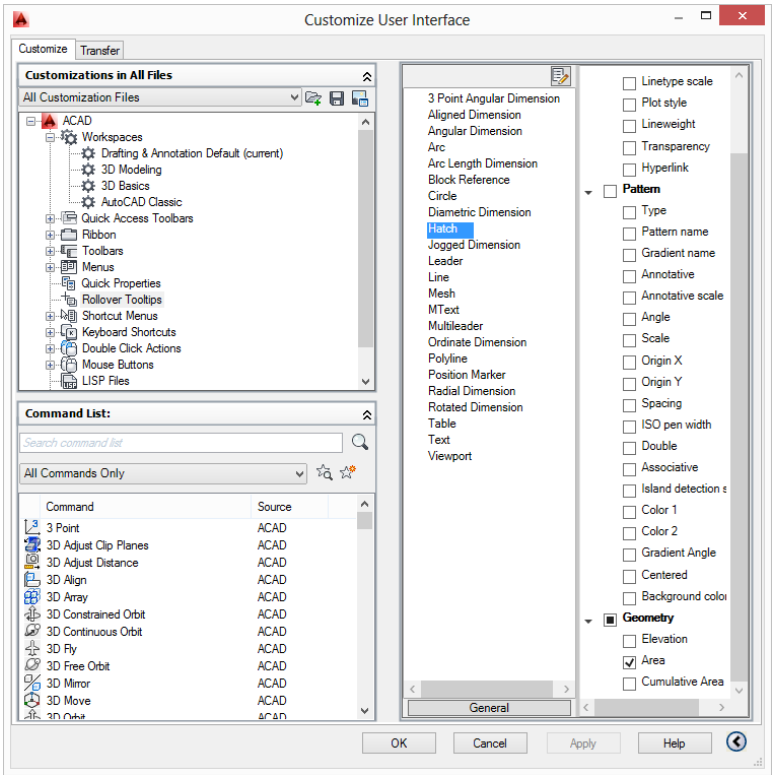


FIGURE 5.30
Modifying the properties displayed on the rollover tooltip for a Hatch object

4. In the Properties list, to the right of the Object Type list, click Area under the Geometry category.
5. Click OK to save the changes made.
6. Create a closed area and apply a hatch pattern to it.
7. Position the cursor over the hatch object, and you should see Area as a property on the rollover tooltip. If you customized the properties for the Quick Properties palette, select the hatch object and right-click. Click Quick Properties to display the Quick Properties palette if it is not already displayed.

TIP You can synchronize the properties set between rollover tooltips and the Quick Properties palette. Right-click over the Quick Properties or Rollover Tooltips nodes in the Customizations In pane of the CUI Editor, and click Synchronize With Rollover Tooltips or Synchronize With Quick properties. In the task dialog box that is displayed, click the link that represents the direction of the synchronization you want to perform.

Organizing the User Interface with Workspaces

Workspaces are used to control which user-interface elements are displayed and where they are positioned in the AutoCAD user interface. You can create and modify a workspace using the CUI Editor or directly manipulate elements in the user interface. If you show/hide user-interface elements or change their position from the AutoCAD user interface, you can save the changes to the current workspace or a new workspace with the `wssave` command. Directly manipulating user-interface elements is often easier than using the CUI Editor, but not all elements can be directly manipulated.

After you create a new element in the CUI Editor, you can control how and where that element is displayed with the Workspace Contents pane. Earlier in this chapter, I explained how to add a ribbon tab, pull-down menu, and toolbar to the current workspace. In earlier sections, I had you working with the current workspace only, but you can modify any workspace in the main customization (CUIx) file, create a new workspace, or even duplicate a workspace.

It is not uncommon to create multiple workspaces. For example, AutoCAD comes with several workspaces that are designed for people who are doing 2D drafting or 3D modeling. You could create separate workspaces for those who work on mechanical, architectural, or civil drawings within your company. You might also encourage users to create their own workspaces with their favorite tools, following your company standards.

These steps explain how to create a new workspace based on the Drafting & Annotation workspace that is defined in the `acad.cuix` file:

1. Display the CUI Editor if it is not open. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the Customizations In pane of the CUI Editor, expand the Workspaces node and select the Drafting & Annotation workspace.
3. Right-click over the Drafting & Annotation workspace node and select Duplicate.
4. In the Properties pane, enter **`Custom Workspace`** in the Name field.

5. Optionally, in the Properties pane, change any of the other available properties for the workspace as needed.
6. In the Workspace Contents pane, click Customize Workspace.
7. In the Customizations In pane, select the user-interface elements you want to display or deselect those you do not want displayed.
8. In the Workspace Contents pane, do any or all of the following, and then click Done:
   - Select the Quick Access Toolbar node and specify if you want to display the QAT above or below the ribbon.
   - Expand the Toolbars node and select the first toolbar. Change the settings in the Properties pane as needed. Do this for each toolbar that is under the Toolbars node.
   - Expand the Menus node and drag to reorder the pull-down menus for the menu bar.
   - Expand the Palettes node and select the first palette. Change the settings in the Properties pane as needed. Do this for each palette that is under the Palettes node.
   - Expand the Ribbon Tabs node and select the first tab. Change the settings in the Properties pane as needed. Do this for each tab that is under the Ribbon Tabs node.
   - Expand the first ribbon tab node under Ribbon Tabs, and select the first panel. Change the settings in the Properties pane as needed. Do this for each panel that is on the selected ribbon tab, and then do this for each tab that is under the Ribbon Tabs node.
9. Click OK to save the changes to the workspace.

You can also change the display state and position of user-interface elements from the AutoCAD user interface and update a workspace by doing the following:

1. With the CUI Editor closed, do any of the following:
   - Quick Access toolbar: Right-click over a command or control, and then click Show Quick Access Toolbar Below/Above The Ribbon to move the QAT above or below the ribbon.
   - Ribbon tabs: Right-click over a tab and click Show Tabs, and then click the tab you want to show/hide that is associated with the current workspace. You can also click and drag a tab on the ribbon to change its display order.
   - Ribbon panels: Click a tab to set it as current and then right-click over the tab. Click Show Panel and then click the panel you want to show/hide that is associated with the tab. You can also click and drag a panel on the ribbon to change its display order, or even drag the panel outside of the ribbon to float the panel. When floating, the panel remains displayed even upon switching tabs.
   - Toolbars: On the ribbon, click View tab ➢ User Interface panel ➢ Toolbars drop-down menu ➢ <*Customization Group Name*> and then the name of the toolbar you want to show or hide. After a toolbar is displayed, drag it on the screen and along the inner edges of the application window to dock it.

- Pull-down menus: On the Quick Access toolbar, click the Customization menu button located on the right side. Click Show/Hide Menu Bar to control the display of the menu bar and pull-down menus. The `menubar` system variable can also be used to display or hide the menu bar. You can't add or remove pull-down menus from the user interface unless you load a partial menu or use AutoLISP or a custom program.
- Palettes: On the ribbon, click View tab ➢ Palettes and click the palettes you want to display as part of the workspace. After a palette is displayed, drag it on screen and along the edges of the application window to dock it.
- Status bar: On the status bar, click the Application Status Bar Menu button and click Drawing Status Bar to control the state of the application and drawing-window status bars. You can also control the state of the status bar with the `statusbar` system variable.

2. On the QAT, from the Workspace drop-down list select Save Current As (or at the command prompt, type **`wssave`** and press Enter).
3. In the Save Workspace dialog box, select a workspace from the drop-down list to update an existing workspace or enter a name to create a new workspace. Click Save. If you selected a workspace name, click Replace.

TIP The process for placing toolbars and palettes just where you want them can be tedious. After you finalize the position, use the Lock Toolbar/Window Positions control on the status bar to lock their position so they are not accidentally moved. You can also use the `lockui` system variable to lock toolbars and palettes in place.

After you have created a workspace, you must set it as current before the changes to the user interface will take place. Use any of the following methods to set a workspace as current:

- Select a workspace from the Workspace drop-down list on the QAT, or choose one from the Workspace Switching icon on the status bar.
- In the Customizations In pane of the CUI Editor, expand the Workspaces node and select the workspace you want to set as current. Right-click the workspace and select Set Current.
- On the Workspace toolbar, select a workspace from the Workspace drop-down list.
- Prior to launching AutoCAD, you can use the `/w` command-line switch with a Desktop shortcut to set a specific workspace as current. For more information on command-line switches, see Chapter 4, "Manipulating the Drawing Environment."
- While AutoCAD is running, you can use the `wscurrent` system variable to see which workspace is current and even switch workspaces.

TIP If you want to retain any changes to the workspace that you make through the user interface before switching to a different workspace, make sure you change the When Switching Workspaces setting in the Workspace Settings dialog box (`wssettings` command). This dialog box also lets you specify the order in which workspaces appear on drop-down lists in the user interface.

Working with Customization Files

Customization (CUIx) files are used to store the definitions of the elements that make up most of the AutoCAD user interface. `acad.cuix` is the default customization file that ships with AutoCAD, and it's the file that you have been customizing throughout this chapter if you're using the default AutoCAD installation. You can create and load additional customization (CUIx) files that contain just your customization with the CUI Editor. I explain how to create and load CUIx files later in the "Creating CUIx Files" and "Loading CUIx Files" sections. When a CUIx file is loaded, you can also load any AutoLISP files that might be required for the command macros defined in the file. I discuss loading AutoLISP files that are related to a CUIx file later in this chapter in the "Loading AutoLISP Files" section.

Earlier AutoCAD releases shipped with files named `acad.mnu` and `acad.cui`, which were used to store the element definitions of the user interface. You can migrate user-interface elements from earlier releases using the Transfer tab of the CUI Editor. I explain how to transfer elements between two customization files in the "Transferring User-Interface Elements between CUIx Files" section.

The AutoCAD user interface supports three types of customization files:

Main The main CUIx file should be writeable and typically contains most of, if not all, of the default AutoCAD user-interface elements. In most configurations, this is the `acad.cuix` file that ships with AutoCAD.

Enterprise The enterprise CUIx file is read-only by design and typically contains your corporate customization, but it could also be the `acad.cuix` file that ships with AutoCAD. When the `acad.cuix` file is designated as the enterprise CUIx file, then your corporate customization is often designated as your main CUIx file. The enterprise CUIx file itself might not be marked as read-only, but the CUI Editor does not allow you to make changes to it.

Partial Partial customization files, as the name implies, do not contain all of the elements that you might find in the standard AutoCAD user interface. The user-interface elements used by third-party utilities and plug-ins, and even the Express Tools user-interface elements, are often implemented with partial CUIx files. These typically contain a few toolbars, ribbon tabs and panels, and pull-down menus, but they can also contain dozens of additional user-interface elements.

No matter whether you use all three types of customization files or just a main CUIx file, consider storing your CUIx file in a centralized location so it can be shared with others. I recommend that you use a partial CUIx file at least for your personal and corporate customization so that you can share it with others in your company, and to make it easier to back up and transition to the latest release.

TIP You can edit the elements of the enterprise CUIx file by setting it as the main customization file and by setting the main customization file as the enterprise CUIx file. To make this easier, you can create two different user profiles that invert the paths for the main and enterprise CUIx files.

Creating CUIx Files

No matter whether you are creating a main, enterprise, or partial CUIx file, the process is exactly the same. The contents and how the file is being loaded are what differentiate one

from another. While you can copy and rename a CUIx file through Windows Explorer or File Explorer, you should avoid doing so because it does not change the customization group name inside the file.

The following steps show how to create a new CUIx file named `myui.cuix`:

1. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the CUI Editor, click the Transfer tab.
3. In the Customizations In pane, on the right click Create A New Customization File.

4. Click Save The Current Customization File.

5. In the Save As dialog box, browse to the folder that you created for this book and type **`myui`** in the File Name text box. Click Save.

If you want to create a new CUIx file from an existing CUIx file, you will want to open the file and then save it with a new name. Use these steps to open and save a CUIx file with a different name:

1. On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface.
2. In the CUI Editor, click the Transfer tab.
3. On the right side of the Customizations In pane, click Open Customization File.

4. In the Open dialog box, browse to and select the CUIx file you want to open. Click Open.
5. Click Save The Current Customization File.
6. In the Save As dialog box, browse to the folder that you want to save the new file to and enter a new name in the File Name text box. Click Save.

Loading CUIx Files

After you create a CUIx file or obtain a CUIx file from a third-party developer or consultant, you must load it so the elements defined in the file can be accessed from the AutoCAD user interface. How you plan to use a CUIx file determines how the file needs to be loaded. You can load a CUIx file using one of the following options:

Main Customization File Click the Application Menu button and then click Options. In the Options dialog box, select the Files tab and expand the Customization Files node. Expand the Main Customization File node and select the path to the CUIx file. Click Browse. In the Select A File dialog box, browse to and select the CUIx file you want to load as the main customization file. Click Open, and then click OK to close the Options dialog box.

Enterprise Customization File Click the Application Menu button and then click Options. In the Options dialog box, select the Files tab and expand the Customization Files node. Expand the Enterprise Customization File node and select the path to the CUIx file. Click Browse. In the Select A File dialog box, browse to and select the CUIx file you want to load as the enterprise customization file. Click Open, and then click OK to close the Options dialog box.

Partial Customization File On the ribbon, click Manage tab ➢ Customization panel ➢ User Interface. In the CUI Editor, select the Customize tab, and in the Customizations In pane, click Load Partial Customization File. In the Open dialog box, browse to and select the CUIx file that

you want to load. Click Open. The CUIx file is added to the Partial Customization Files node of the main customization file.

Transferring User-Interface Elements between CUIx Files

The CUI Editor not only allows you to create and modify elements for the user interface and manage CUIx files, but also lets you copy or migrate elements between two CUIx files or an earlier menu source (MNS) or customization (CUI) file. You transfer elements between files using the Transfer tab. Load the files that you want to work with, and then drag elements between the available nodes in the Customization In panes. Transferring an element also transfers any associated commands, ribbon panels, and toolbars. When transferring elements, you can only drag and drop elements of the same type. So, there's no dragging ribbon tabs to a Quick Access toolbar (QAT).

Loading AutoLISP Files

It is not uncommon that the commands in your CUIx files might use functions or commands defined in an AutoLISP file. There are a few different methods that you can use to make sure that the AutoLISP programs your user-interface elements rely on are loaded for use by your CUIx file. The following outlines the methods you can use to load an AutoLISP file for use with a CUIx file:

- AutoCAD searches for an AutoLISP Menu (MNL) file that has the same name as a CUIx file that is being loaded. If the MNL file is located, AutoCAD loads it along with the CUIx file into each drawing that is created or opened while the CUIx file is loaded.
- Using the CUI Editor, you can add AutoLISP (LSP) files to the LISP Files node. These files are loaded when the CUIx file is loaded and a new drawing is created or opened.
- Using the Load/Unload Applications dialog box (`appload` command), you can manually load an AutoLISP (LSP) file or add it to the Startup Suite. The Startup Suite loads the files that are listed when a new drawing is created or an existing drawing file is opened.

Controlling the Tools on the Status Bars

AutoCAD supports two status bars: drawing and application window. By default, both are combined into a single bar and displayed along the bottom of the application window. While you can control which tools are displayed, you can't create and place new tools on the status bar. Even though you can't add new tools, it can be helpful to hide those that you do not use, such as 3D Object Snap and Allow/Disallow Dynamic UCS if you do not work on 3D models.

Unlike other user-interface elements, the settings that control the display of the tools on the toolbar are not stored in the CUIx file or controlled by the current workspace, with the exception of the current status-bar state (`statusbar` system variable). To control which tools are displayed on the status bar, do one of the following:

Application Status Bar Menu Click the Application Status Bar Menu button (see Figure 5.31) to control the display of the coordinates area and drafting aids on the status bar. You can also access controls related to layers, drawing views, workspaces, and display locking, among others. This menu allows you to display the Tray Settings dialog box and toggle the display of the drawing status bar.

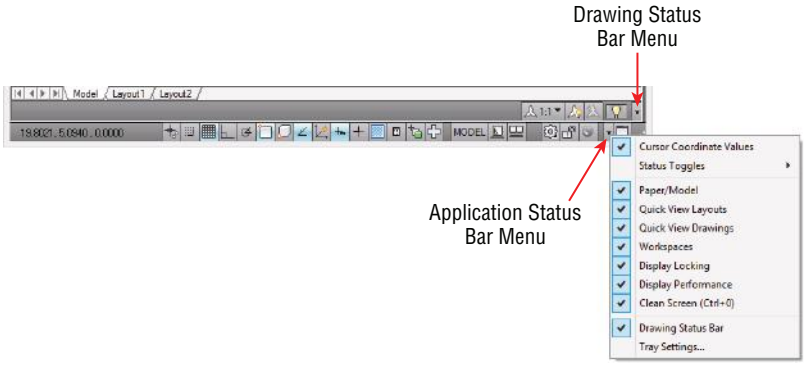


FIGURE 5.31
Controlling the display of tools on the status bar

Drawing Status Bar Menu When the drawing status bar is displayed, it contains the controls related to annotation scaling and the system tray. Click the drawing status-bar menu to toggle which annotation scaling tools you want to display.

Tray Settings Dialog Box Controls the display of icons and notifications for services that are running in the application or current drawing. You can also specify the duration that a notification is displayed for, or whether it is displayed until you close it. The actual enabling or disabling of a service is not handled in this dialog box; you must do that on a feature-by-feature basis. For example, you can use the `xrefnotify` system variable to control the display of xref notifications or the `layernotify` system variable to display alerts for unreconciled new layers.

Chapter 6

Customizing the AutoCAD User Interface for Mac

The Autodesk® AutoCAD® user interface provides you with one of the best opportunities to increase your productivity without learning a programming language. Many of the user-interface elements that allow you to start a command or toggle a system variable with a click of a button can be customized. By customizing the user interface, you can reorganize it to better fit the way you work or add the commands that you frequently use, and even remove those that you do not.

You customize the user interface with the Customize dialog box (`cui` command). As you make changes to the user interface, the changes are stored as part of the AutoCAD property list (Plist) files or in the main customization (CUI) file. CUI files contain the definitions for buttons and menu items that are displayed on the Tool Sets palette and the menus displayed on the menu bar.

Getting Started with the Customize Dialog Box

The Customize dialog box, shown in Figure 6.1, is the tool that you will need to become familiar with if you plan to customize the AutoCAD user interface. The Customize dialog box is a simpler interface to learn and use than its Windows counterpart, the Customize User Interface (CUI) Editor. The Customize dialog box in AutoCAD on Mac OS is similar to the Customize dialog box that was in AutoCAD on Windows prior to AutoCAD 2006.

You can display the Customize dialog box using one of the following methods:

- Click Tools ➢ Customize ➢ Interface (CUI).
- At the command prompt, type **`cui`** and press Enter.

When the Customize dialog box is displayed, notice the three tabs along the top. Each of these tabs allows you to perform a specific task related to customizing the user interface. The three tabs in the Customize dialog box are as follows:

Commands Tab Use the Commands tab to create, modify, and remove the commands that can be placed on a toolset to create a button or a pull-down menu to create a menu item.

Menus Tab Use the Menus tab to create, modify, remove, and organize the pull-down menus that are displayed on the menu bar in the user interface. You can add or remove commands to or from a pull-down menu, as well as create submenus and insert separators to organize similar commands.

Tool Sets Tab Use the Tool Sets tab to create, modify, remove, and organize the toolsets that are displayed as part of the Tool Sets palette. You add or remove commands to or from a tool

group that's used to display commands on a toolset. Drop-downs can also be used to organize similar commands.

NOTE Click the Reset To Defaults button to reset the CUI files to their original installed state. When you do so, all customization to the pull-down menus and toolsets will be removed, along with any new commands or changes to existing commands that you might have made.

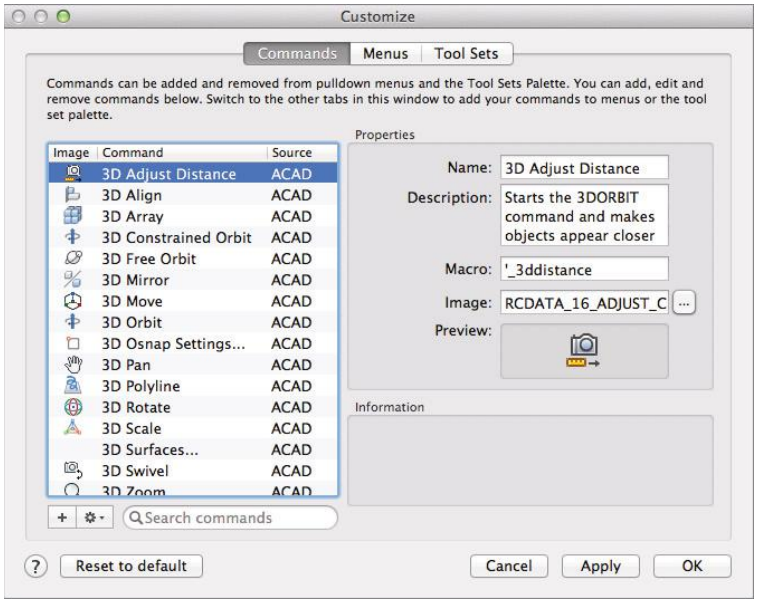


FIGURE 6.1 You create and modify user-interface elements by using the Customize dialog box.

Creating Commands and Defining Command Macros

Commands are the primary component of an element in the AutoCAD user interface, and they are created on the Commands tab of the Customize dialog box. The properties of a command in a CUI file define the sequence of AutoCAD commands and options that should be executed when the command is used, as well as how that command should appear on a user-interface element. The sequence of AutoCAD commands and options that a command passes to AutoCAD is known as a *macro*. The macro is the most significant property of a command in a CUI file.

Understanding the Basics of a Command Macro

A command macro defines the input that should be sent to the AutoCAD command prompt when a user-interface element is used; as with the standard command macros that come with AutoCAD, you can see that you do not need to start and complete a command as part of a command macro. You can start a command with one macro and then click another user-interface element to send an expected value to the active command. For example, one macro might start

a custom AutoLISP® routine that prompts you for a bolt or window size; instead of typing a size each time, you could use a second macro to pass a value to the routine.

For the most part, a command macro is similar to the input that you enter at the command prompt to start and complete an AutoCAD command, but it can also contain special characters that control its execution. For example, the following might be what you normally would do to draw a circle with a diameter of 1/8":

```
Command: CIRCLE

Specify center point for circle or [3P/2P/Ttr (tan tan radius)]: Specify a point
with the pointing device

Specify radius of circle or [Diameter] <0.1875>: d

Specify diameter of circle <0.3750>: 0.125
```

A command macro performing the same task and using the same input might look something like this:

```
^C^C._circle;\_d;0.125;
```

As you can see, I used five different special characters in the example macro that were not present as part of the original input entered at the command prompt: ^ (caret), . (period), _ (underscore), ; (semicolon), and \ (backslash). Table 6.1 explains the significance of each character used in the example macro.

TABLE 6.1: Macro components used in the circle example

MACRO COMPONENT	DESCRIPTION
^C^C	Simulates the pressing of the Esc key twice.
._circle	Passes the `circle` command to the command prompt.
;	Simulates the pressing of the Enter key to start the `circle` command. (Could also be a single space in the macro.)
\	Pauses for the user to specify the center point of the circle.
_d	Indicates that the Diameter option of the `circle` command will be used.
;	Simulates the pressing of the Enter key to accept the Diameter option.
0.125	Specifies the value for the Diameter option.
;	Simulates the pressing of the Enter key to accept the Diameter value. Since this is the last expected value, the `circle` command ends too.

Table 6.2 lists the most common special characters used in a command macro.

TABLE 6.2: Special characters that can be used in macros

SPECIAL CHARACTER	DESCRIPTION
`^C`	Equivalent to pressing Esc.
`;`	Equivalent to pressing Enter.
`[blank space]`	Equivalent to pressing Enter or spacebar based on the expected input of the current prompt.
`\`	Allows the user to provide input.
`.`	Accesses an AutoCAD command's standard definition even when a command might have been undefined with the `undefine` command.
`_`	Instructs AutoCAD to use the global command name or option value instead of the localized name or value provided. This allows the macro to function as expected when used with a different language of the AutoCAD release.
`*`	Repeats the AutoCAD command after the asterisk character until the user cancels the command. Example macro from the Point, Multiple Point command in the `menugroup.cui` file: `*^C^C_point`
`$M=`	Indicates the start of a DIESEL expression. Example expression from the UCS Icon, On command in the `menugroup.cui` file: `$M=$(if,$(and,$(getvar,ucsicon),1),^C^C_ucsicon _off,^C^C_ucsicon _on)`

NOTE You can learn about other special and control characters that can be used in a command macro by searching the AutoCAD Help system using the keywords *characters in macros*.

The next example walks you through the process of converting input entered at the command prompt into a command macro. You will make a layer, set that layer as current, and then add a multiline text (MText) object.

1. At the command prompt, type **`-layer`** and press Enter.
2. At the `Enter an option` prompt, type **`m`** and press Enter.
3. At the `Enter name for new layer (becomes the current layer) <0>:` prompt, type **`Notes`** and press Enter.
4. At the `Enter an option` prompt, type **`c`** and press Enter.

5. At the `New color [Truecolor/COlorbook]:` prompt, type **8** and press Enter.
6. At the `Enter name list of layer(s) for color 8 <Notes>:` prompt, press Enter.
7. At the `Enter an option` prompt, press Enter.
8. At the command prompt, type **-mtext** and press Enter.
9. At the `Specify first corner:` prompt, specify a point in the drawing area.
10. At the `Specify opposite corner or [Height/Justify/Line spacing/Rotation/Style/Width/Columns]:` prompt, type **j** and press Enter.
11. At the `Enter justification [TL/TC/TR/ML/MC/MR/BL/BC/BR] <TL>:` prompt, type **tl** and press Enter.
12. At the `Specify opposite corner or [Height/Justify/Line spacing/Rotation/Style/Width/Columns]:` prompt, type **h** and press Enter.
13. At the `Specify height <0.2000>:` prompt, type **0.25** and press Enter.
14. At the `Specify opposite corner or [Height/Justify/Line spacing/Rotation/Style/Width/Columns]:` prompt, type **r** and press Enter.
15. At the `Specify rotation angle <0>:` prompt, type **0** and press Enter.
16. At the `Specify opposite corner or [Height/Justify/Line spacing/Rotation/Style/Width/Columns]:` prompt, type **w** and press Enter.
17. At the `Specify width:` prompt, type **7.5** and press Enter.
18. At the `MText:` prompt, type **NOTE: ADA requires a minimum turn radius of** and press Enter.
19. At the `MText:` prompt, type **60" (1525mm) for wheelchairs.** and press Enter.
20. Press Enter again to end the `mtext` command.
21. Press the FN-F2 key combination to expand the command-line history.
22. In the History area, select the command prompts that were displayed and the input that you entered (see Figure 6.2). Secondary-click (right-click or two-finger-click) and click Copy.
23. In the Mac OS Finder, click Go ➢ Applications. In the Finder window, double-click TextEdit.
24. In TextEdit, click in the editor window and press Command-V to paste the copied text from the command-line history.
25. From the pasted text, remove the two informational lines `Current layer:` and `Current text style:` and their values.
26. Replace the `Specify first corner:` prompt with a \ (single backslash character).
27. Remove all the other prompts before the input you entered.

28. After each line, with the exception of the line that contains the backslash, add a ; (semicolon). After this and the previous three steps you should have the following left in TextEdit.

```
-layer;
m;
Notes;
c;
8;
;
;
-mtext;
\
j;
tl;
h;
0.25;
r;
0;
w;
7.5;
NOTE: ADA requires a minimum turn radius of;
60" (1525mm) diameter for wheelchairs.;
;
```

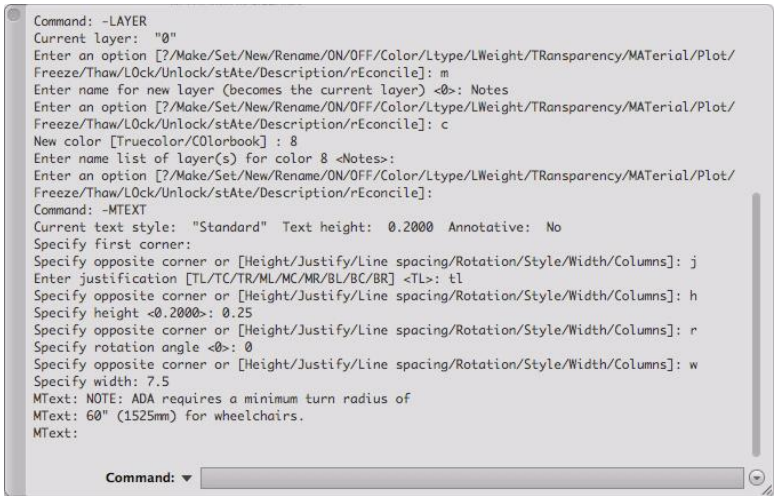


FIGURE 6.2 Command-line history of the input you previously entered

29. Enter **^C^C** before the first line to make sure no other command is active when the macro is used.

30. Place the cursor at the end of each line and press Delete to move all input to a single line. Your finished macro should look like this:

```
^C^C-layer;m;Notes;c;8;;;-mtext;\j;tl;h;0.25;r;0;w;7.5;NOTE: ↵
ADA requires a minimum turn radius of;60" (1525mm) for wheelchairs.;;
```

31. Do not close TextEdit, as you will use the macro you created in the next section to create a command for use in the user interface.

TIP Add **._** in front of each command name and an **_** (underscore) in front of the values that represent an option name. This will ensure your macro works correctly if a command is undefined or the command macro is used on a non-English AutoCAD release.

Creating and Modifying Commands

Now that you understand how to define a command macro, you can create a command that will allow you to use the macro from the AutoCAD user interface. Commands are created from the Commands tab of the Customize dialog box. In addition to creating new commands, you can locate and modify existing ones.

The following steps show how to create a command for the wheelchair note macro that you created in the previous section. If you did not complete the previous exercise, you can open the `NoteMacro.txt` file available for download from this book's web page at `www.sybex.com/go/autocadcustomization`.

1. Click Tools ➢ Customize ➢ Interface (CUI) (or at the command prompt, type **cui** and press Enter).

2. When the Customize dialog box opens, on the Commands tab under the Command list, click the + (plus sign) to create a new command.

3. In the Properties section, type **Wheelchair Note** in the Name field. The Name field is used to identify the command in the Command list and is part of the tooltip that is displayed when the cursor is over the command if placed in a tool group as part of a toolset.

4. In the Macro field, type the macro that you created in the previous exercise or copy/paste the contents of the `NoteMacro.txt` file into the text box. The macro defines the actions that AutoCAD will perform when the command is used from the user interface.

5. Optionally, in the Description field type **ADA Wheelchair minimum radius note**. The text entered in this field helps to describe what the command is used for and is part of the tooltip that is displayed when the cursor is over the command when placed on a toolset.

6. Click Apply to save the changes made to the new command's properties (see Figure 6.3).

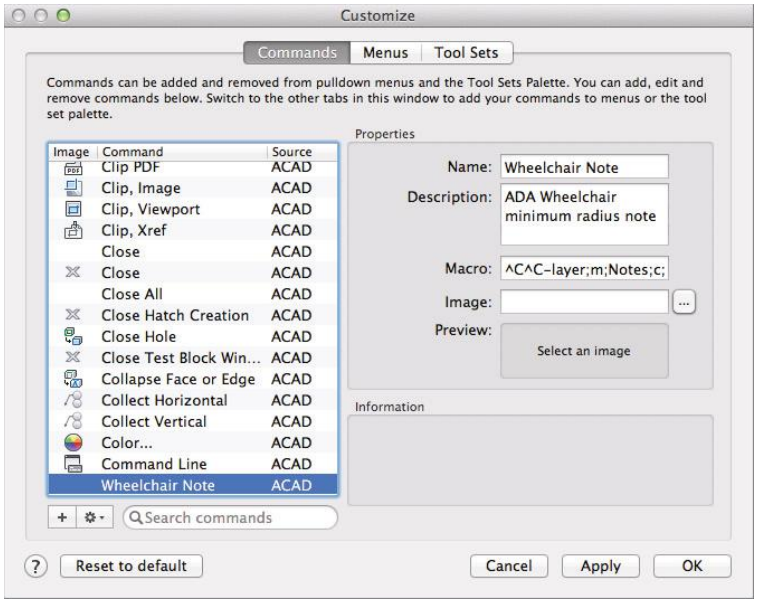


FIGURE 6.3
The properties of the completed command

A command can be edited by selecting it from the Command list and then making changes to it in the Properties section, just as you did when you created the new command.

Assigning an Image to a Command

While using images with your commands is optional, you should consider adding an image to each of the commands that you create using the Customize dialog box to provide the most flexibility as to where your commands are used in the user interface. There is no internal image editor in AutoCAD for Mac, so you will need to use an external application to create the images.

Here are the basic requirements:

- Images should be 16 × 16 pixels in size; the program will scale them up as needed. Using a higher resolution such as 32 × 32 will look better on higher-resolution displays.
- Images can be of the ICO, ICNS, PNG, JPEG, TIFF, BMP, RLE, or DIB file formats.

The following steps show how to assign a custom image to the Wheelchair Note command that you created in the previous section:

1. Display the Customize dialog box if it is not currently open. Click Tools ➤ Customize ➤ Interface (CUI).
2. In the Customize dialog box, on the Commands tab select the Wheelchair Note command. The Command list is sorted alphabetically; you can type **wheelchair** in the Search Commands text box to help locate the command faster.
3. Under the Properties section, click the ellipsis […] button to the right of the Image field.

4. In the Select An Image File dialog box, browse to the files that you downloaded from the book's web page and select the file named `WheelchairNote.bmp`. Click Open.
5. Click Apply to save the changes to the command. The Customize dialog box and the Wheelchair Note command should now look like Figure 6.4.

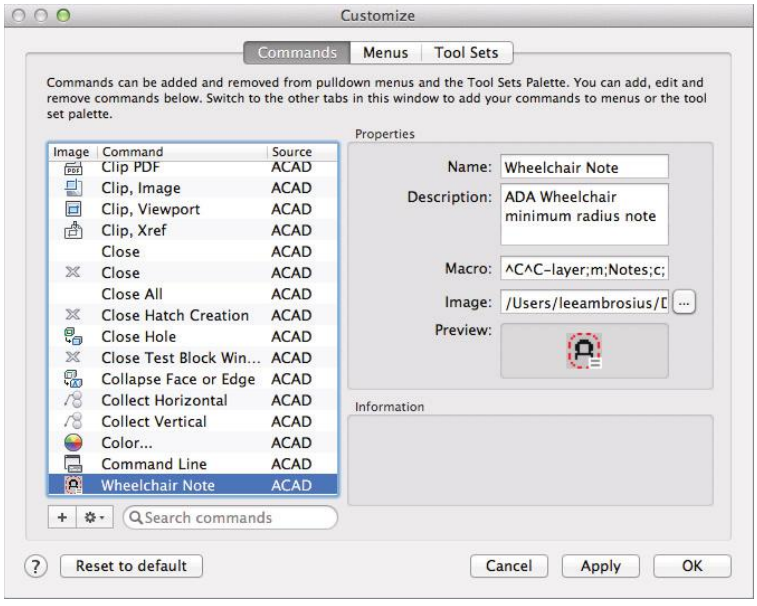


FIGURE 6.4 Wheelchair command with the custom icon

NOTE I recommend that you store your custom images in the location specified by the Custom Icon Files node under Customization Files in the Application Preferences dialog box (`options` command).

Customizing User-Interface Elements

Out of the box, the AutoCAD user interface is designed for everyone, but not for any specific industry or any one single company's workflow. You can add new elements that execute command macros and custom applications you create, remove those that you do not use, or reorganize those that you use frequently to make them easier to access.

The following elements of the user interface can be created and edited with the Customize dialog box:

- Pull-down menus
- Toolsets and groups; Tool Sets palette

Pull-Down Menus

The menu bar is an area found along the top of the screen and displays a series of menus—also referred to as *pull-down menus*—related to the active application. You click one of the labels on the menu bar to display the associated pull-down menu. When the pull-down menu is displayed, you click items to start an associated command macro. A pull-down menu can contain separators and submenus to group related commands together. The right side of the menu bar can contain additional items that are not controlled by the active application but are related to systemwide settings, such as the clock or battery level; none of these items can be customized.

Pull-down menus contain the commands defined on the Commands tab of the Customize dialog box. The command macros that you define can use DIESEL (Direct Interpretively Evaluated String Expression Language) to make decisions with conditional statements, and you can also use DIESEL as part of the menu item's name after you add a command to the pull-down menu to display a check box next to a menu item. For information on what DIESEL is and how to use it, refer to the AutoCAD Help system.

The following exercise shows how to create a new pull-down menu, add multiple commands to it, and organize the commands with a separator and submenu:

1. Display the Customize dialog box if it is not currently open. Click Tools ➢ Customize ➢ Interface (CUI).
2. When the Customize dialog box opens, click the Menus tab.
3. On the Menus tab, in the Menus list on the right side, select Parametric. This item represents the Parametric pull-down menu that appears on the menu bar.
4. Click the + (plus sign) button ➢ Add Menu to create a new pull-down menu after the Parametric pull-down menu. Once added, you can drag a pull-down menu in the Menus list to change its display order on the menu bar.
5. In the in-place editor, type **Favorites** for the name of the new pull-down menu and press Enter.
6. In the Commands list, locate and add the Wheelchair Note command to the Favorites pull-down menu. Type **wheelchair** in the Search Command text box. Click and drag the Wheelchair Note command to the Favorites pull-down menu under the Menus list. When the cursor is over Favorites, release the mouse button.

NOTE Once a command is added to the pull-down menu, you can change its display name by changing the Display Name field in the Menus list. This is the field you need to edit if you want to include a DIESEL string to display a check mark next to a menu item, similar to the items on the Palettes submenu on the Tools pull-down menu when a palette is displayed.

7. Click the arrow (disclosure triangle) next to the Favorites pull-down menu to expand it.
8. Select the Wheelchair Note command that you added, and then click the + (plus sign) button ➢ Add Sub-menu to create a new submenu below the Wheelchair Note command. In the in-place editor, type **Annotation Tools** for the name of the new submenu and press Enter.

9. In the Commands list, locate and add the Multiline Text and Single Line Text commands to the Annotation Tools submenu. Type **line text** in the Search Command text box to make it easier to find the two commands.
10. In the Menus list, select the Wheelchair Note command, and secondary-click (right-click or two-finger-click) over the command. Click Insert Separator.
11. Click and drag the Annotation Tools submenu above the separator.
12. In the Command list, locate and add the Multileader command below the separator. Type **multileader** in the Search Command text box to make it easier to find the command. Figure 6.5 shows what the completed pull-down menu should look like under the Menus list.

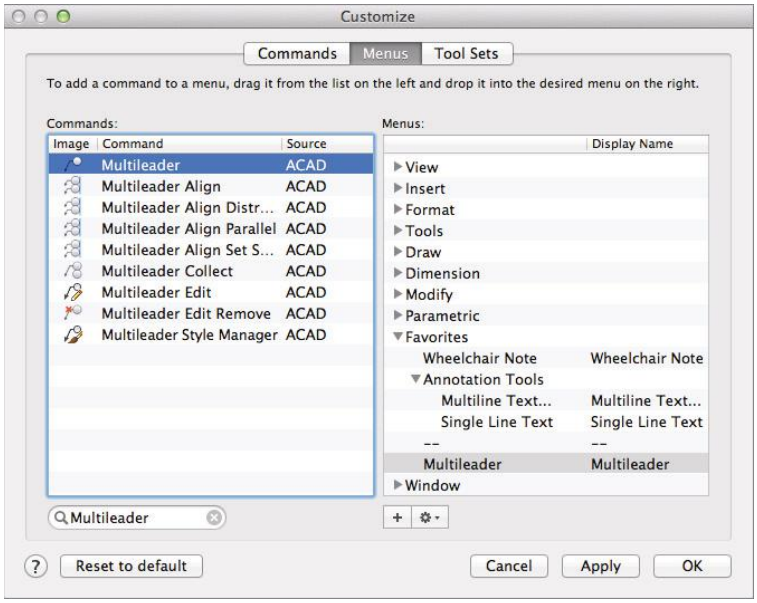


FIGURE 6.5 Structure of the Favorites pull-down menu in the Customize dialog box

13. Click OK to save the new pull-down menu.
14. On the menu bar, click Favorites. Figure 6.6 shows what the Favorites pull-down menu should look like on the menu bar. Click the Wheelchair Note item and then specify a point in the drawing to test the command and its macro. Figure 6.7 shows the results of the command macro. Continue to test the other menu items and submenu of the pull-down menu.

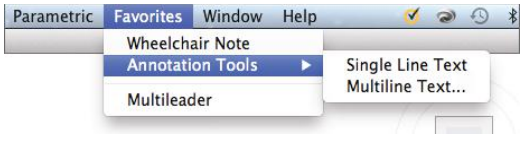


FIGURE 6.6 Favorites pull-down menu on the menu bar

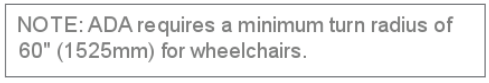


FIGURE 6.7
The results of using the Wheelchair Note command

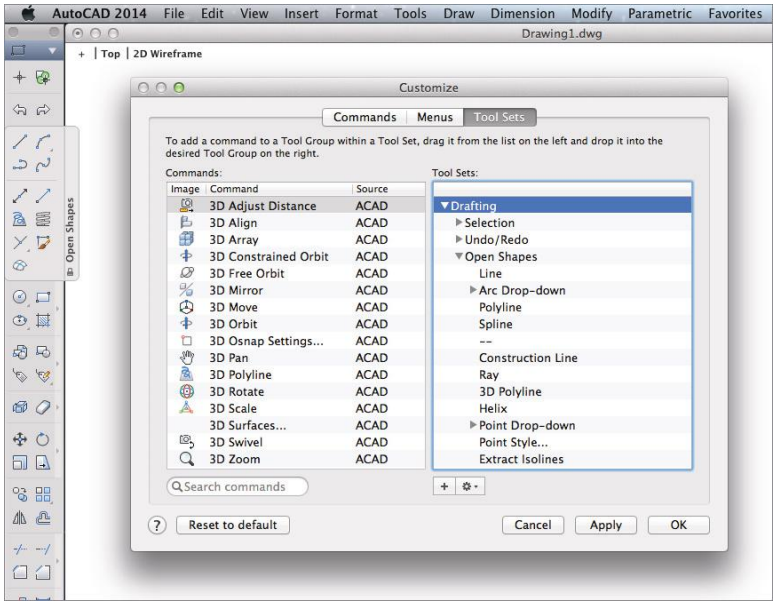
Toolsets

The Tool Sets palette is the visual user interface on AutoCAD for Mac that allows you to start a command. It is a cross between toolbars and the ribbon found in AutoCAD for Windows or Microsoft Office 2013. Near the top of the Tool Sets palette is the Tool Set button, which allows you to switch between the currently displayed toolset and other available toolsets. Three default toolsets come with the program, and these are based on the tasks of 2D drafting, 3D modeling, and annotating a drawing.

Each toolset can be made up of one or more tool groups, which allow you to group similar commands. Tool groups can be in one of two different display states: normal or expanded. When defining a tool group, you can specify which commands are displayed in the normal state of the tool group by placing them above a separator. Those added below the separator are displayed when the tool group is expanded.

If a tool group contains commands below the separator, an arrow is displayed along the left or right side of the tool group when it is displayed on the Tool Sets palette. Clicking the arrow expands the group so you can use the commands that are not displayed by default. When a group is expanded, it can be locked to remain expanded until you unlock it. Drop-downs can be added to a tool group to further organize related commands. Figure 6.8 shows the Open Shapes tool group in the Drafting toolset on the Tool Sets palette and how it is defined in the Customize dialog box.

FIGURE 6.8
Structure of the Open Shapes tool group



The following explains how to create a new toolset named Favorites and a group named My Tools. Once the My Tools group is created, you will add several commands to it, including the Wheelchair Note command that you created earlier in this chapter.

1. Display the Customize dialog box if it is not currently open. Click Tools ➤ Customize ➤ Interface (CUI).
2. When the Customize dialog box opens, click the Tool Sets tab.
3. On the Tool Sets tab, in the Tool Sets list on the right, select Modeling. This item represents the Modeling toolset that is accessible from the Tool Sets palette.
4. Click the + (plus sign) button ➤ Add Tool Set to create a new toolset. Once added, you can drag a toolset in the Tool Sets list to change its order on the menu that is displayed when the Tool Sets button is clicked at the top of the Tool Sets palette.
5. In the in-place editor, type **Favorites** for the name of the new toolset and press Enter.
6. Click the arrow (disclosure triangle) next to the Favorites toolset.
7. Click the default tool group under the Favorites toolset, and then click the tool group a second time to display the in-place editor to rename the tool group. Type **My Tools** for the name of the tool group and press Enter. Click the + (plus sign) button ➤ Add Tool Group to add other tool groups if desired.
8. Click the arrow (disclosure triangle) next to the My Tools tool group.
9. Select the My Tools tool group and click the + (plus sign) button ➤ Add Drop-Down to add a drop-down menu to the tool group. Your toolset and tool group should now look like Figure 6.9.

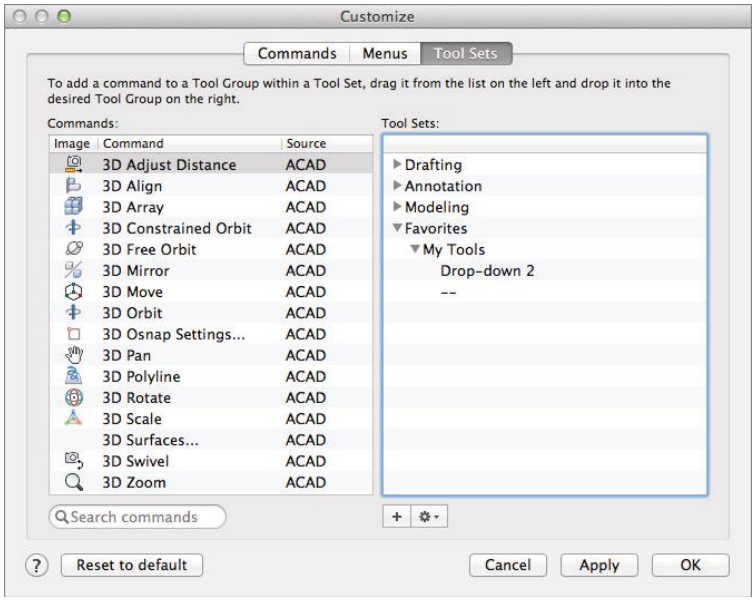


FIGURE 6.9 Structure for the new Favorites toolset and the My Tools tool group

10. Locate the Wheelchair Note command in the Command list and add it to the My Tools tool group above the new drop-down. Type **Wheelchair** in the Search Command text box. Click and drag the Wheelchair Note command to the My Tools tool group. When the cursor is below the My Tools tool group and above the drop-down menu, release the mouse button.
11. In the Command list, locate and add the Multileader, Multileader Edit, and Multileader Edit Remove commands to the drop-down. Remember, you can use the Search Command text box to filter the Command list.
12. Add the Multileader Style Manager command between the Wheelchair Note and drop-down.
13. Add the other five multileader-related commands under the separator, which is represented by the – – item under the My Tools tool group.
14. Create a new tool group and add some of your favorite commands, if you want. Figure 6.10 shows the new toolset and group in the Customize dialog box.

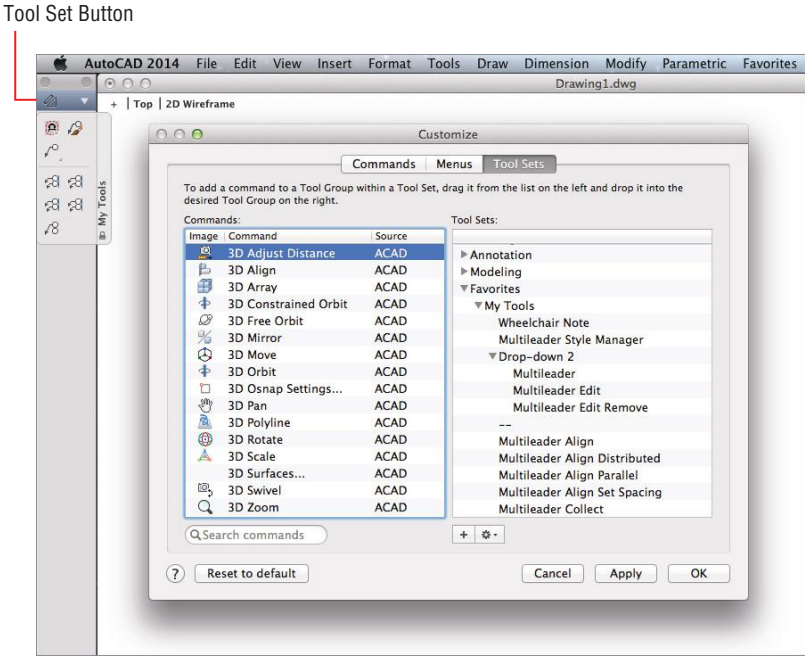


FIGURE 6.10 Completed Favorites toolset and My Tools tool group

15. Click OK to save the new toolset.
16. On the Tool Sets palette, click the Tool Sets button near the top of the palette and click Favorites. Test the tools on the tool group and expand the tool group to see the commands placed below the separator.

NOTE Secondary-click (right-click or two-finger-click) over a toolset in the Tool Sets list of the Customize dialog box to assign an image to it. This image is displayed on the toolset menu when the Tool Sets button is clicked.

Controlling the Tools on the Status Bar

AutoCAD displays a status bar that allows you to access 2D-drafting and 3D-modeling aids, switch layouts, and set the scale to use for annotation scaling and the current viewport, among other options. While you can control which tools are displayed, you can't create and place new tools on the status bar. Even though you can't add new tools, it can be helpful to hide those that you do not use, such as, say, Polar Tracking and ViewCube®. You can also choose to hide the status bar by clicking Tools ➢ Palettes ➢ Status Bar or using the `statusbar` system variable.

Unlike other user interface elements, the settings that control the display of the tools on the status bar are not stored in the main customization (CUI) file; instead, they are stored with the program's property list (Plist) files. When controlling which tools are displayed on the status bar, you use the following options:

3D Status Bar Disclosure Triangle Click the Show/Hide 3D Status Bar disclosure triangle located on the right side of the status bar to show or hide the 3D-related tools. The 3D-related tools are displayed above the standard tools on the status bar.

Status Bar Context Menu The status bar's context menu (see Figure 6.11) allows you to toggle which drafting aids, layout, annotation scaling, viewport, and other available tools are displayed or hidden on the status bar. Which items are displayed on the menu is dependent on whether or not the 3D status bar is shown. Secondary-click (right-click or two-finger-click) in an empty area of the status bar to display its context menu.

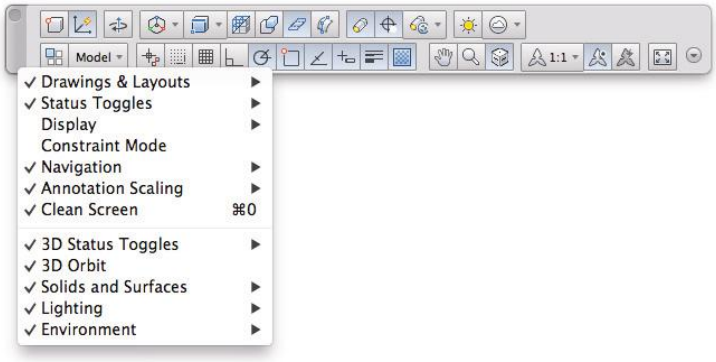


FIGURE 6.11 Controlling the display of tools on the status bar

Chapter 7

Creating Tools and Tool Palettes

The Autodesk® AutoCAD® user interface has changed and evolved over the years to make it easier to access your favorite tools and allow you to create new custom tools. First introduced with AutoCAD 2004, tool palettes allow you to create tools dynamically, on the fly, based on the objects you create in a drawing or the files stored on a local/shared drive. AutoCAD comes with a number of predefined tool palettes that show the wide range of tools that a tool palette can contain.

Tools are organized on a tool palette and displayed using the Tool Palettes window. The Tool Palettes window is a modeless window, or dockable palette, with tabs along its left or right side; each tab represents one of the available tool palettes. Tool palettes can be organized into groups, allowing you to display only the tools you need based on the current task you are performing, and they can also be shared with other users.

NOTE Tool palettes are available in AutoCAD on Windows only.

What Is a Tool Palette?

A tool palette is a container for user-defined content, and each is stored in a file with the extension `.atc`. ATC files are based on the Extensible Markup Language (XML). You create and edit these files when customizing tool palettes from the AutoCAD user interface with the Tool Palettes window or Customize dialog box.

TIP ATC files can also be edited with caution from outside of AutoCAD using either an ASCII text editor, such as Notepad, or a specialized editor that is designed for XML files. Editing ATC files outside of AutoCAD can make it easy to update the paths used to reference several content types, such as blocks, raster images, and external references.

Figure 7.1 shows what several of the default tool palettes look like in the Tool Palettes window.

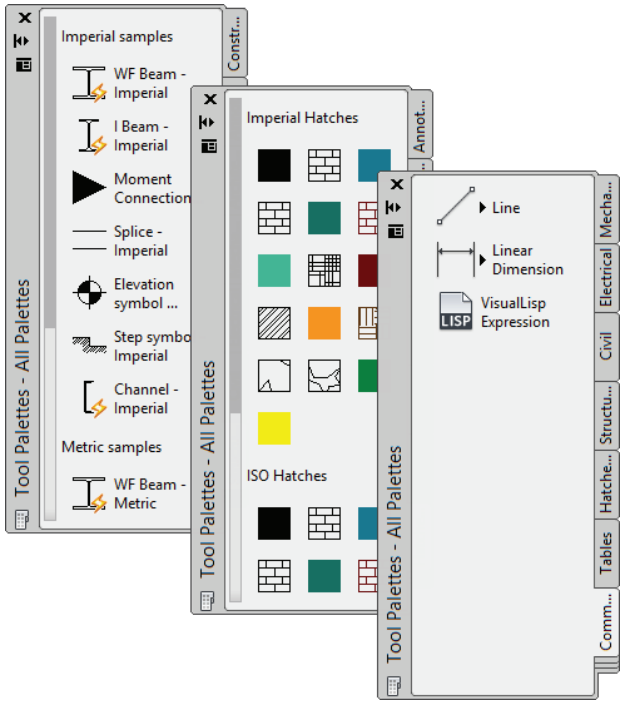


FIGURE 7.1 Tool palettes in the Tool Palettes window

Each tool palette can contain a number of different tools that are based on geometry in a drawing or an external file. A tool can also execute a command macro, an action macro, or an AutoLISP® expression. Once a tool is added to a tool palette, you can customize its properties. Most of the tools support two classifications of properties: those specific to the tool's type and those that are general to most objects in a drawing, such as color, linetype, and lineweight.

Using the Tool Palettes Window

The Tool Palettes window is the interface that is used to access the tools and tool palettes that come with AutoCAD, as well as those that you create yourself. Some of the functionality of the window should have a familiar feel to it for a couple of reasons. First, it shares a common framework with other palettes in AutoCAD, such as the Properties palette and the Layer Properties Manager. Second, tools can be displayed using Icon and Detail views, similar to the way files can be displayed in Windows Explorer or File Explorer.

You can display the Tool Palettes window using one of the following methods:

- On the ribbon, click View tab ➢ Palettes panel ➢ Tool Palettes.
- At the command prompt, type **`toolpalettes`** or **`tp`** and press Enter.

Figure 7.2 shows the main components of the Tool Palettes window. An explanation of each component follows.

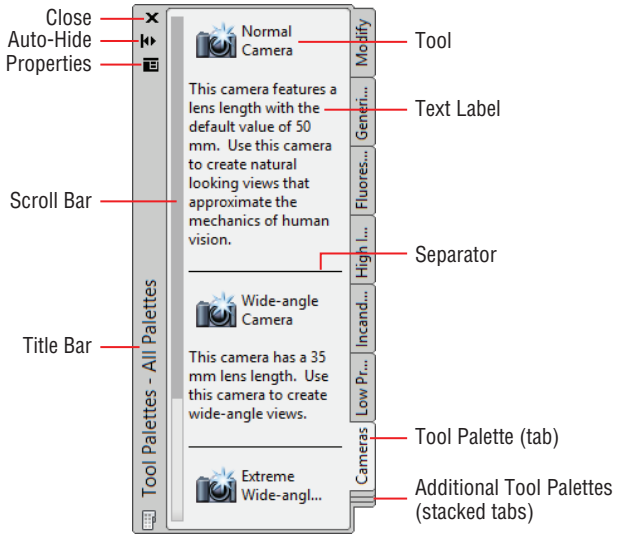


FIGURE 7.2 Components of the Tool Palettes window

Close Button Click the Close button to close the Tool Palettes window.

Tool Button A tool button represents one of the available tools on the current tool palette. The icon displayed for each tool varies by tool type. Click a button, or click and drag a tool into the drawing area to activate the tool. Right-click a button to display a menu that allows you to edit the tool.

Tool Button with Flyout When there is a flyout, the tool button represents a tool with more than one option; these are available on geometry- and dimension-related tools. Click the arrow next to the tool to display the additional tools.

Text Label A text label provides additional information about the tools on a tool palette.

Separator A separator is a horizontal bar that allows you to visually group related tools on a tool palette.

Tool Palette (Tab) Each tab represents one of the ATC files found in one of the designated locations in the Options dialog box (`options` command). Click a tab to switch tool palettes. Right-click a tab or in an empty area of a tool palette to customize it.

Additional Tool Palettes (Stacked Tabs) Stacked tabs indicate that there are additional tool palettes besides those currently visible as tabs in the Tool Palettes window. Click the stack of tabs to display a menu with all available tool palettes; select a tool palette from the list to set it as current.

Scroll Bar The scroll bar is displayed when there are tools on the current tool palette that are not visible. Click or drag the scroll bar to see the tools that are not currently visible. You can also click and drag in an empty area of the tool palette to scroll the listing of tools.

Title Bar The title bar displays the name of the palette and current tool-palette group. Click and drag the title bar to move the Tool Palettes window, or right-click it to display a menu to anchor or customize the Tool Palettes window.

TIP You can resize the Tool Palettes window by positioning the cursor along its top or bottom edge. When the double-sided arrow cursor appears, click and drag to change the height and/or width of the palette. You can also enable resizing mode by right-clicking over the title bar and clicking Size. When the cursor changes to a multidirection cursor, click and drag over the Tool Palettes window to resize the palette.

Auto-Hide Toggle The Auto-Hide control collapses the Tool Palettes window down to its title bar only, resulting in more available screen real estate than when the window is not collapsed and docked. Passing the cursor over the title bar expands the Tool Palettes window temporarily.

Properties Button The Properties button displays a menu of options that allow you to anchor or customize the Tool Palettes window, create a new tool palette, add new command tools, or set a tool-palette group as current, among other tasks.

Defining Tool Palettes

Now that you are familiar with the Tool Palettes window and how it works, it's time to dig in and take a closer look at how to customize tool palettes. As mentioned earlier, tool palettes are the containers used to store user-defined content—also known as *tools*. Most of the options that you will need to create and manage tool palettes are accessed from the Tool Palettes window.

Creating a Tool Palette

When creating your own tools, I recommend not using any of the default tool palettes that come with AutoCAD, and instead creating your own tool palettes. When you create your own tool palettes, you make it easier to share your tools with others and migrate your tools forward to a new release.

The following steps explain how to create a new tool palette called My Tools:

1. On the ribbon, click View tab ➢ Palettes panel ➢ Tool Palettes (or at the command prompt, type **`toolpalettes`** and press Enter).
2. In the Tool Palettes window, right-click a tool palette tab or in an empty area of the current tool palette. Click New Palette.
3. In the in-place editor, type **My Tools** and press Enter. The new tab is set as current and is ready for you to add tools to it.

New tool palettes are created in the first writable folder AutoCAD finds under the Tool Palettes File Locations node on the Files tab of the Options dialog box (`options` command). You can modify the paths AutoCAD uses to locate tool palette files by using the `*_toolpalettepath` system variable.

Modifying Tool Palettes

After a tool palette has been created, you can add tools to or modify the palette. I discuss adding tools later in this chapter in the section "Adding and Modifying Tools." You can perform the following tasks to modify a tool palette:

- Rename a tool palette
- Delete a tool palette that is no longer needed
- Reorder a tool palette in the Tool Palettes window
- Control the view style for the tools on a tool palette

Renaming a Tool Palette

You can rename a tool palette by doing the following:

1. In the Tool Palettes window, right-click the tool palette's tab you want to rename and click Rename Palette.
2. In the in-place editor, type a new name and press Enter.

Deleting a Tool Palette

A tool palette can be removed by following these steps when it is no longer needed:

1. In the Tool Palettes window, right-click the tool palette's tab you want to remove and click Delete Palette.
2. In the Confirm Palette Deletion message box, click OK to remove the palette.

TIP If you think you might want to restore a tool palette later, export the tool palette before removing it. I discuss how to export and import a tool palette in the section "Sharing Tool Palettes and Tool-Palette Groups" later in this chapter.

Reordering a Tool Palette in the Tool Palettes window

You can adjust the order in which a tool palette is displayed in the Tool Palettes window by following these steps:

1. In the Tool Palettes window, right-click the tool palette's tab you want to change the order of.
2. Click Move Up or Move Down.

Controlling the View of Tools on a Tool Palette

The image size and view style of the tools on a tool palette can be adjusted by doing the following:

1. In the Tool Palettes window, right-click the tool palette's tab whose view options you want to change and click View Options.
2. When the View Options dialog box (see Figure 7.3) opens, change the Image Size and View Style options.

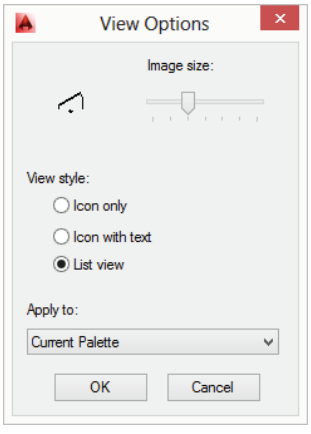


Figure 7.3 Controlling the image size and style for tools

3. From the Apply To drop-down list, select Current Palette to change the view options for the current palette only. Select All Palettes to apply the change to all of the palettes available from the Tool Palettes window (not just those in the current tool-palette group).
4. Click OK.

Adding and Modifying Tools

Tools are the main component of any tool palette. For the most part, the tools on a tool palette are based on the same objects that you would add to a drawing. These objects can be both graphical and nongraphical. Some types of tools can execute a command macro or use files that are stored externally of a drawing file. Once a tool is added to a tool palette, you can change its properties to align with your established CAD standards.

The types of tools that you can add to a tool palette are as follows:

- Geometry
- Dimensions, leaders, and geometric tolerances
- Annotation tools (text, tables, and wipeouts)
- Blocks (static and dynamic)
- Hatch-pattern and gradient fills

- External references and raster images
- Lights
- Visual styles
- Materials
- Cameras
- Commands

NOTE If you are using an AutoCAD-based product, such as AutoCAD® Architecture, you might have access to additional types of tools that can be added to a tool palette.

Creating a Tool

The tool-creation process that you need to follow depends on the type of tool you want to create. As I previously mentioned, most of the source information used to create a tool comes from the objects in a drawing, but in some cases the information also comes from external files that are stored outside of a drawing.

The following steps show how to create a Geometric tool based on a graphical object within a drawing file:

1. Create a new drawing file.
2. Create a new layer named `Objs` and assign it the color `Red (1)`.
3. Create a second new layer named `Lines` and assign it the color `Green (3)`.
4. Set the `Objs` layer as current.
5. On the `Objs` layer, draw a line object using the `line` command.
6. On the ribbon, click View tab ➢ Palettes panel ➢ Tool Palettes.
7. Click the My Tools tab to set it as current. (If you do not have a My Tools tab in the Tool Palettes window, go back to the section "Creating a Tool Palette" earlier in the chapter and follow the steps to create the tool palette.)
8. Select the line object that you drew in step 5. Press and hold down the right pointer button, and drag the line object over the My Tools tool palette in the Tool Palettes window (see Figure 7.4). Do not click over one of the grips. Release the pointer button to create the tool (see Figure 7.5).

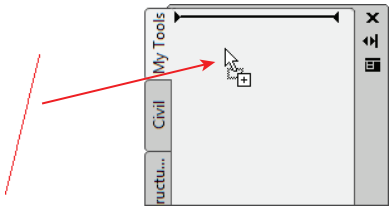


FIGURE 7.4 Adding a new tool based on a line object from the drawing area

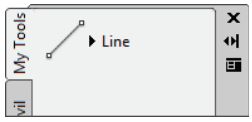


FIGURE 7.5
New Line tool on a custom tool palette

9. Set layer `0` as current.
10. In the Tool Palettes window, click the new Line tool. The `line` command starts; specify a start and endpoint for the command; then end the command. The new line object is added to the `Objs` layer even though layer `0` was current.
11. Click the arrow on the right side of the Line tool. On the flyout (see Figure 7.6), click the Circle tool. The `circle` command is started and any new circles you create with the tool are placed on the `Objs` layer.

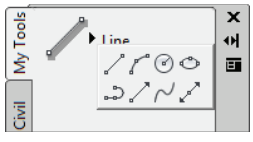


FIGURE 7.6
A flyout offers additional related geometric tools.

12. Right-click the Line tool and click Properties.
13. When the Tool Properties dialog box (see Figure 7.7) is opened, you can edit the tool's properties. In the General section, select the Layer property and then click the drop-down list that is displayed. Select `Lines` from the list of available layers in the current drawing.

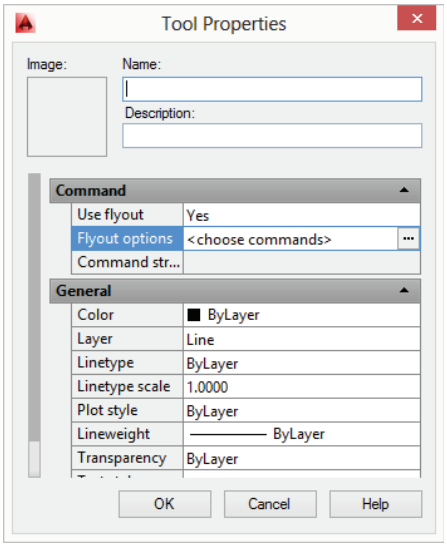


FIGURE 7.7
Changing the properties of a tool

14. In the Command section, select the Flyout Options property and then click the ellipsis […] button. In the Flyout Options dialog box (see Figure 7.8), clear all check boxes except Line, Arc, and Polyline. Click OK.

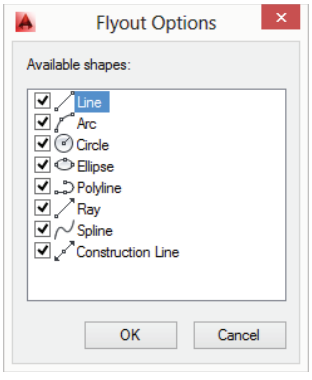


Figure 7.8 Controlling which tools are displayed on the flyout

15. Click OK again to save the changes to the tool and close the Tool Properties dialog box.
16. Try the Line tool again. Any lines drawn with the tool now are placed on the `Lines` layer.
17. Click the arrow on the right side of the Line tool. The only tools that are displayed on the flyout this time are Line, Arc, and Polyline (see Figure 7.9).

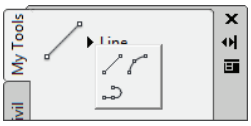


Figure 7.9 Revised tool and flyout

Geometric Tool

You can create a Geometric tool based on objects in one of your drawings by doing the following:

1. Display the Tool Palettes window.
2. Select one of the following objects from the current drawing:
   - Line
   - Arc
   - Circle
   - Ellipse
   - Polyline
   - Ray
   - Spline
   - Xline (Construction line)
3. Drag and drop the object onto a tool palette.

4. Right-click the new tool and click Properties. Edit the properties in the Command section. Change the Use Flyout property to No if you wish to disable the flyout for the tool, or change which tools are displayed on the flyout by editing the Flyout Options property. Click OK to save the changes to the tool.

Dimension Tool

You can create a Dimension tool based on the dimension objects in one of your drawings by following these steps:

1. Display the Tool Palettes window.
2. Select one of the following objects from the current drawing:
   - Aligned dimension
   - Linear dimension
   - Arc Length dimension
   - Jogged dimension
   - Radius dimension
   - Diameter dimension
   - Angular dimension
   - Ordinate dimension
   - Quick leader
   - Geometric tolerance
3. Drag and drop the object onto a tool palette.
4. Right-click the new tool and click Properties. Edit the properties in the Command section. Change the Use Flyout property to No if you wish to disable the flyout for the tool, or change which tools are displayed on the flyout by editing the Flyout Options property. Click OK to save the changes to the tool.

Multileader Tool

You can create a Multileader tool based on existing multileader objects in one of your drawings by doing the following:

1. Display the Tool Palettes window.
2. Select a multileader object in the current drawing.
3. Drag and drop the object onto a tool palette.

Annotation Tools

You can create an Annotation tool in one of your drawings with these steps:

1. Display the Tool Palettes window.
2. Select one of the following objects in the current drawing:
   - Single-line text
   - Multiline text

- Table
- Wipeout

3. Drag and drop the object onto a tool palette.

Hatch Tool

You can create a Hatch tool based on a hatch pattern or gradient fill object in one of your drawings by doing the following:

1. Display the Tool Palettes window.
2. Select a hatch pattern or gradient fill object in the current drawing.
3. Drag and drop the object onto a tool palette.

A Hatch tool can also be created from a hatch pattern in a PAT (pattern image) file by doing the following:

1. Display the Tool Palettes window.
2. On the ribbon, click View tab ➢ Palettes panel ➢ DesignCenter (`adcenter` command).
3. On the DesignCenter™ palette, click the Search button along the top.
4. When the Search dialog box (see Figure 7.10) opens, click the Look For drop-down list and select Hatch Pattern Files.

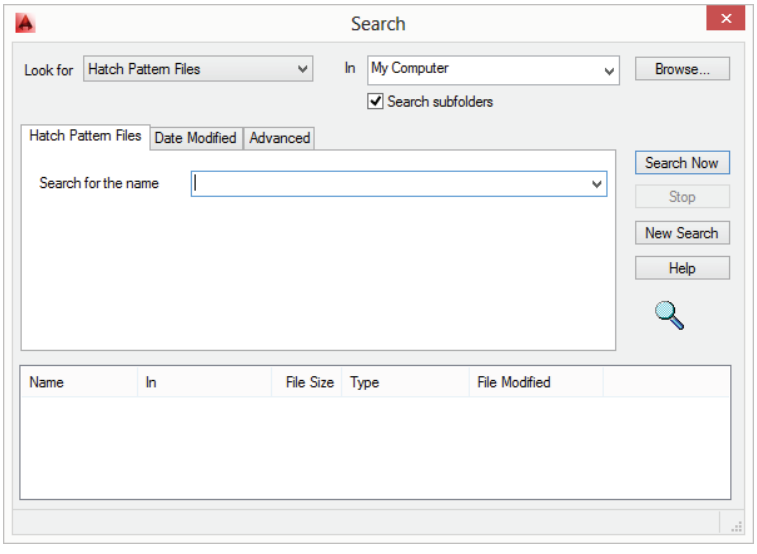


Figure 7.10 Searching for named objects and support files

5. Click the In drop-down list and select My Computer.

6. On the Hatch Pattern Files tab, in the Search For The Name text box, type ***.pat** and click Search Now.
7. In the Results list, double-click the file `acad.pat`. The `acad.pat` file is loaded into DesignCenter, and a thumbnail is displayed for each hatch pattern present in the file.
8. Drag and drop a hatch-pattern thumbnail from DesignCenter onto a tool palette. (If you want to create a gradient fill, you can create a Hatch tool based on any pattern and then change the tool's Tool Type property from Hatch to Gradient with the Tool Properties dialog box.)

Block Tool

You can create a Block tool based on a block reference object inserted into one of your drawings by doing the following:

1. Save the current drawing.
2. Display the Tool Palettes window.
3. Select a block reference object in the current drawing.
4. Drag and drop the object onto a tool palette.

You can also create a Block tool from a DWG file:

1. Display the Tool Palettes window.
2. Open Windows Explorer (Windows XP and Windows 7) or File Explorer (Windows 8), and browse to the drawing (DWG) file you want to create a Block tool from.
3. In Windows Explorer or File Explorer, click and drag the drawing file and drop it onto a tool palette.

As an additional alternative, a Block tool can also be created with DesignCenter by doing the following:

1. Display the Tool Palettes window.
2. On the ribbon, click View tab ➢ Palettes panel ➢ DesignCenter (`adcenter` command).
3. On the DesignCenter palette, browse to the drawing (DWG) file or drawing file that contains the block definition you want to create a Block tool from.
4. Do one of the following:
   - Click and drag the drawing file, and drop it onto a tool palette to create a Block tool based on all the objects contained in the drawing file's model space.
   - Double-click the drawing file to see all the named objects within it. Double-click the Blocks item to view all blocks in the drawing file. Click and drag a block definition, and drop it onto a tool palette to create a Block tool.

NOTE The definition of the block does not become part of the Block tool, but the tool maintains information about where the block definition exists: the drawing file and the block name.

External Reference or Raster Image Tool

You can create an External Reference (xref) or Raster Image tool based on an xref or raster image attached to a drawing by doing the following:

1. Save the current drawing.
2. Display the Tool Palettes window.
3. Select an xref or raster image object in the current drawing.
4. Drag and drop the object onto a tool palette.

You can also create an xref or Raster Image tool from a DWG or supported image file by doing the following:

1. Display the Tool Palettes window.
2. Open Windows Explorer (Windows XP and Windows 7) or File Explorer (Windows 8), and browse to the drawing or supported image file you want to create a tool from.
3. In Windows Explorer or File Explorer, click and drag the file, and drop it onto a tool palette.
4. If you created the tool by dragging and dropping a DWG file, right-click the tool and click Properties.
5. When the Tool Properties dialog box opens, select the Insert As property. Click the drop-down list and select Xref. Click OK.

As an additional alternative, an xref or Raster Image tool can be created with DesignCenter by doing the following:

1. Display the Tool Palettes window.
2. On the ribbon, click View tab ➢ Palettes panel ➢ DesignCenter (`adcenter` command).
3. On the DesignCenter palette, browse to the drawing or supported image file you want to create a tool from.
4. Click and drag the file, and drop it onto a tool palette.

3D Visualization Tools

You can create a Light or Camera tool based on a light or camera object in one of your drawings with these steps:

1. Display the Tool Palettes window.
2. Select a light or camera object in the current drawing.

3. Drag and drop the object onto a tool palette.

The following explains how to create a Visual Style tool based on a saved visual style in a drawing:

1. Display the Tool Palettes window.
2. On the ribbon, click View tab ➢ Palettes panel ➢ Visual Styles Manager (`visualstyles` command).
3. On the Visual Styles Manager, click and drag one of the thumbnails along the top of the window, and drop it onto a tool palette.

You can create a Material tool based on a material stored in one of your drawings or your library by doing the following:

1. Display the Tool Palettes window.
2. On the ribbon, click View tab ➢ Palettes panel ➢ Materials Browser (`matbrowseropen` command).
3. In the Materials Browser, click and drag one of the materials listed in the Document Materials or current library list, and drop it onto a tool palette.

Command Tool

You can create a Command tool that executes a command macro by following these steps:

1. Display the Tool Palettes window.
2. In the Tool Palettes window, right-click in an empty area of the current tool palette or the Tool Palettes window's title bar. Click Customize Commands.
3. In the Customize User Interface (CUI) Editor, from the Command List pane, click and drag one of the commands and drop it onto a tool palette. I explain how to work with the CUI Editor and define commands in Chapter 5, "Customizing the AutoCAD User Interface for Windows."

Using a Tool

A tool can be started with a single left click, but some can also be started by being dragged and dropped into the drawing area. When supported, the two actions often invoke slightly different responses. For example, if you click a Hatch tool, you are prompted for an insertion point, which is used to identify the closed boundary that should be filled. If you drag a Hatch tool into the drawing area and drop it into a closed boundary, you are not prompted to specify an insertion point or to click inside a closed boundary. When you drop the Hatch tool, AutoCAD uses the current position of the cursor as the point it should try to use to create the hatch-pattern fill.

Modifying Tools

After a tool has been created, you can manage the tool on a tool palette or change its properties. You can perform the following tasks to modify a tool:

- Change a tool's properties
- Specify a new image for a tool
- Delete a tool that is no longer needed
- Copy or duplicate a tool
- Move a tool on a tool palette or to a different tool palette
- Sort tools on a tool palette

Changing the Properties and Image of a Tool

After a tool has been created, you can change its properties and image:

1. In the Tool Palettes window, right-click the tool you wish to change and click Properties.
2. In the Tool Properties dialog box, do any of the following:
   - Edit the values of the properties for the tool, enter a value in the text boxes, or select a value from the drop-down lists.
   - Right-click over the image to the left of the Name and Description text boxes and click Specify Image. In the Select Image File dialog box, browse to and select the image file you want to use.
3. Click OK to save the changes you made.

TIP You can also right-click over a tool and click Specify Image to choose a new image for the tool.

Deleting a Tool

When a tool is no longer needed, you can remove it:

1. In the Tool Palettes window, right-click the tool you wish to remove and click Delete.
2. In the AutoCAD message box, click OK to remove the tool.

TIP If you think you might want to restore the tool later, export the tool palette that the tool is on before removing it. I discuss how to export a tool palette in the section "Sharing Tool Palettes and Tool-Palette Groups" later in this chapter.

Copying/Duplicating a Tool

If you want to create a new tool that is similar to a tool that already exists on a tool palette, do the following:

1. In the Tool Palettes window, right-click the tool to duplicate and click Copy.
2. Optionally, click a tool palette tab to add the copied tool to a different tool palette.
3. Right-click in an empty area of the tool palette and click Paste.
4. Click and drag the tool to reposition it on the tool palette.

Moving a Tool

You can change the position of a tool on the current palette or move it to a different palette by doing one of the following:

- In the Tool Palettes window, click and drag a tool to move it on the current palette. Release the pointer button when the tool is where you want to place it.
- In the Tool Palettes window, right-click the tool you want to move to a different palette and click Cut. Click one of the other tool-palette tabs. Right-click in an empty area of the tool palette and click Paste. Click and drag the tool to reposition it on the tool palette.

Sorting Tools

The tools on a tool palette can be sorted by type or name by doing the following:

1. In the Tool Palettes window, right-click in an empty area of the tool palette.
2. Click Sort ➢ Name or Type.

Adding Text Labels or Separators

Text labels and separators allow you to provide information about the tools on a tool palette and to group related tools together. You can create either of these by doing the following:

1. Display the Tool Palettes window.
2. In the Tool Palettes window, right-click in an empty area of the current tool palette and click Add Text or Add Separator.
3. If you clicked Add Text, type a text message in the in-place editor that is displayed and press Enter.
4. Click and drag the new item on the tool palette to change its current placement as needed.

Organizing Tool Palettes with Tool-Palette Groups

Tool-palette groups allow you to control which tool palettes are displayed in the Tool Palettes window. By default, all tool palettes are displayed. You can tell which tool-palette group is active because its name is displayed on the Tool Palettes window's title bar. When all tool palettes are displayed, the title bar displays the text "Tool Palettes - All Palettes."

You can switch tool-palette groups by clicking the Properties button in the Tool Palettes window (see Figure 7.2) and selecting the name of the tool-palette group to make it current. The available tool-palette groups are listed at the bottom of the menu. For the tool palettes you create, consider adding your own tool-palette group; this ensures that only your tool palettes are visible when using the Tool Palettes window.

In the following steps, you'll create a new tool-palette group named My Group and add tool palettes to it:

1. On the ribbon, click Manage tab ➢ Customization panel ➢ Tool Palettes (or at the command prompt, type **`customize`** and press Enter).
2. When the Customize dialog box (see Figure 7.11) opens, scroll to the bottom of the tree view on the right side.

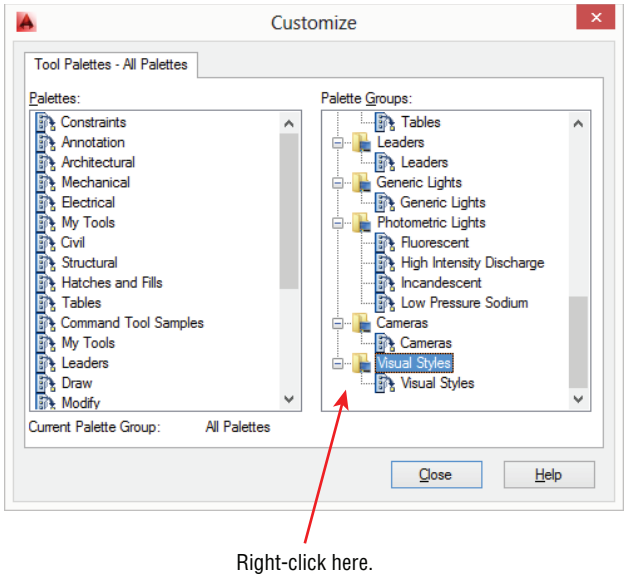


FIGURE 7.11 Creating and managing tool-palette groups

3. Right-click near the bottom of the tree view on the right side, as shown in Figure 7.11. Click New Group. (Do not right-click over one of the existing groups or palettes in a group; that creates a nested group. Right-clicking to the right or left of an item in the tree view does not work either.)
4. In the in-place editor, type **My Group** and press Enter.
5. In the Palettes list, on the left click and drag the My Tools tool palette, and drop it on the My Group tool-palette group.
6. Right-click the My Group tool-palette group and click Set Current.
7. Click Close to save the changes and exit the Customize dialog box.

You can also perform the following tasks from the Customize dialog box:

- Rename a tool palette
- Create a new tool palette

- Delete a tool palette that is no longer needed
- Export and import tool palettes
- Rename a tool-palette group
- Delete a tool-palette group
- Reorder a tool palette within a tool-palette group
- Export and import tool-palette groups

Sharing Tool Palettes and Tool-Palette Groups

Tool palettes by default are stored and created on the local drive of your workstation. It is possible to share the tool palettes you create by doing one of two things:

- Create your tool palettes in a shared location by adding that location as the first writable folder AutoCAD finds under the Tool Palettes File Locations node on the Files tab of the Options dialog box (`options` command). By creating tool palettes in a shared location and having all users point to that location, you ensure that users will not need to import the tool palettes on their workstations.
- Export tool palettes to XTP (Exported Tool Palettes) files with the Customize dialog box (`customize` command), and then import them on another workstation with the Customize dialog box. In the Customize dialog box, from the Palettes list select a tool palette and right-click. Then click Export or Import, and specify or select an XTP file based on the option you choose.

NOTE Autodesk does not recommend sharing tool palettes with users on older releases or across different products. Doing so could result in some tools not working properly—or at all.

Tool-palette groups also are stored on your local drive of your workstation as part of your AutoCAD user profile. You can share with others any tool-palette groups that you create by exporting them from one workstation and importing them onto another. Tool-palette groups can be exported or imported using the Customize dialog box. In the Customize dialog box, in the Palette Groups tree view select a tool-palette group and right-click. Then click Export or Import, and specify or select an XPG file based on the option you chose.

NOTE Before you import a tool-palette group, you must import all tool palettes that are referenced by the group.

Chapter 8

Automating Repetitive Tasks

In Chapters 5 through 7, I discussed how to create custom elements and tools that could be accessed from the user interface. These elements and tools make it easy to execute command macros, which are often designed to start and pass values to the current command.

While a command macro can help you combine multiple commands into a single action, two features of the Autodesk® AutoCAD® program are specifically designed to help reduce and automate repetitive tasks: scripts and action macros. You'll learn how to work with both scripts and action macros in this chapter.

NOTE Action macros and the Action Recorder are supported in AutoCAD on Windows only.

Creating and Running Scripts

Scripts are one of the oldest methods that you can use to automate repetitive tasks in AutoCAD. For example, scripts can open, save, and close a drawing file—an AutoLISP® program can't. You can use scripts for a number of different purposes; here are a few of the most common uses for script files:

- Update the CAD standards in one or more drawings
- Change the settings and nongraphical objects in the drawings that you receive from a client
- Execute a series of commands with specific values in a drawing

The following list outlines the known limitations of scripts:

- Unlike command macros, scripts do not support a way to pause execution and prompt a user for input directly. You can use AutoLISP expressions to request input for a command.
- Dialog boxes should be avoided with scripts. Starting a command that displays a dialog box suspends the script that is executing when the dialog box opens and then resumes the script when the dialog box is dismissed. The problem with displaying a dialog box is that the responses a user provides might not produce the results that were originally intended when the script was written.
- AutoCAD can execute only a single script file at a time, so a script file can't start the `script` command to execute another script.

What Is a Script?

A script file is an ASCII text file that can contain AutoCAD commands, values, and AutoLISP expressions. You can create a script file using a text editor, such as Notepad on Windows or TextEdit on Mac OS. After you create a script file, you must save it with the ANSI encoding and the `.scr` file extension.

The content of a script file mimics the commands and values that you would enter at the AutoCAD command prompt. For example, the following command sequence controls some of the settings in model space, creates a new layer, and then adds a rectangle and a single-line text object:

```
Command: grid
Specify grid spacing(X) or [ON/OFF/Snap/Major/aDaptive/Limits/Follow/Aspect]
   <0.5000>: off
Command: ltscale
Enter new linetype scale factor <1.0000>: 48
Regenerating model.
Command: -layer
Current layer:  "0"
Enter an option [?/Make/Set/New/Rename/ON/OFF/Color/Ltype/LWeight/TRansparency/
   MATerial/Plot/Freeze/Thaw/LOck/Unlock/stAte/Description/rEconcile]: new
Enter name list for new layer(s): border
Enter an option [?/Make/Set/New/Rename/ON/OFF/Color/Ltype/LWeight/TRansparency/
   MATerial/Plot/Freeze/Thaw/LOck/Unlock/stAte/Description/rEconcile]: color
New color [Truecolor/COlorbook] : 9
Enter name list of layer(s) for color 9 <0>: border
Enter an option [?/Make/Set/New/Rename/ON/OFF/Color/Ltype/LWeight/TRansparency/
   MATerial/Plot/Freeze/Thaw/LOck/Unlock/stAte/Description/rEconcile]:
Command: rectang
Specify first corner point or [Chamfer/Elevation/Fillet/Thickness/Width]: 0,0
Specify other corner point or [Area/Dimensions/Rotation]: 1056,816
Command: -text
Current text style:  "Standard"  Text height:  0.2000  Annotative:  No
   Justify:  Left
Specify start point of text or [Justify/Style]: 20,20
Specify height <0.2000>: 12
Specify rotation angle of text <0>: 0
Enter text: This is a sample startup script.
Command: zoom
Specify corner of window, enter a scale factor (nX or nXP), or
[All/Center/Dynamic/Extents/Previous/Scale/Window/Object] <real time>: extents
Regenerating model.
```

An example script containing the same commands and values as those entered at the command prompt might look something like Listing 8.1.

Listing 8.1: Startup Script example

```
; Created 10/1/2013 by Lee Ambrosius - Startup Script
grid off
ltscale 48
-layer new border color 9 border set border
rectang 0,0 1056,816
-text 20,20 12 0 This is a sample startup script.
zoom extents
```

NOTE You can find the Startup Script example script in the file `startup.scr` that is part of the sample files for this book. The sample files can be downloaded from `www.sybex.com/go/autocadcustomization`.

Special Characters in a Script

You should be aware of a few special characters when it comes to writing scripts. These characters include the following:

Semicolon (;) A semicolon denotes a comment; everything to the right of the semicolon is ignored. The first line of the Startup Script example shows what a comment looks like in a script. You can use comments to identify who created a script or when a script was created, as well as the purpose of the script.

Space A space in a script has two different meanings; it can represent either the press of the Enter or Return key or an actual space in a text string. The action a space represents is the same as it would be if you pressed the spacebar at the AutoCAD command prompt.

Carriage Return/Hard Return A carriage return is added when the Enter or Return key is pressed. It is just like pressing Enter or Return when typing commands and values at the AutoCAD command prompt.

NOTE A script needs to end with a space or hard return to process the last value of the last line.

As an alternative, the commands and values in a script can be placed on separate lines. Listing 8.2 shows what the Alternative Startup Script example might look like if each command and value were placed on separate lines.

Listing 8.2: Alternative Startup Script example

```
; Created 10/1/2013 by Lee Ambrosius - Alternative Startup Script
grid
off
ltscale
```

```
48
-layer
new
border
color
9
border
set
border

rectang
0,0
1056,816
-text
20,20
12
0
This is a sample startup script.
zoom
extents
```

NOTE You can find the Alternative Startup Script example in the file `setup_alt.scr` in the sample files for this book. The sample files can be downloaded from `www.sybex.com/go/autocadcustomization`.

Alternatives to Dialog Boxes

As I mentioned earlier, you should avoid the use of commands that display dialog boxes with scripts. Table 8.1 and Table 8.2 show many of the commands that start dialog boxes, as well as the commands or system variables that you should use instead when writing scripts.

Table 8.1: Command-line equivalent commands (Windows and Mac OS)

Command	Command-Line Equivalent	Command	Command-Line Equivalent
array	-array	bhatch	-bhatch
attach	-attach	block	-block
attdef	-attdef	boundary	-boundary
attedit	-attedit	color	-color
bedit	-bedit	cvrebuild	-cvrebuild

TABLE 8.1: Command-line equivalent commands (Windows and Mac OS) *(CONTINUED)*

COMMAND	COMMAND-LINE EQUIVALENT	COMMAND	COMMAND-LINE EQUIVALENT
ddptype	pdmode, pdsize	diminspect	-diminspect
dimstyle	-dimstyle	dsettings	autosnap, dynmode, dyn-prompt, grid, isoplane, ortho, snap, snaptype
etransmit	-etransmit	export	-export
group	-group	hatch	-hatch
hatchedit	-hatchedit	image	-image
imageadjust	-imageadjust	insert	-insert
interfere	-interfere	layer	-layer
linetype	-linetype	lweight	-lweight
mtext	-mtext	objectscale	-objectscale
open	filedia, saveas	opensheet-set	-opensheetset
osnap	-osnap	overkill	-overkill
pan	-pan	plot	-plot
plotstamp	-plotstamp	plotstyle	-plotstyle
properties	celtscale, celtype, change, chprop, clayer, -color, elev, -layer, -linetype, thickness	psetupin	-psetupin
refedit	-refedit	rename	-rename
render	-render	saveas	filedia, saveas
scalelistedit	-scalelistedit	style	-style
text	-text	units	-units
vports	-vports	wblock	-wblock
xbind	-xbind	xref	-xref

TABLE 8.2: Command-line equivalent commands (Windows only)

COMMAND	COMMAND-LINE EQUIVALENT	COMMAND	COMMAND-LINE EQUIVALENT
3dconfig	-3dconfig, -graphicsconfig	actusermessage	-actusermessage
actstop	-actsop	attext	-attext
archive	-archive	dataextraction	-dataextraction
copytolayer	-copytolayer	dgnadjust	-dgnadjust
ddvpoint	vpoint	dgnexport	-dgnexport
dgnattach	-dgnattach	dwfadjust	-dwfadjust
dgnimport	-dgnimport	eattext	-eattext
dwfattach	-dwfattach	fbximport	-fbximport
fbxexport	-fbxexport	inputsearchoptions	-inputsearchoptions
hyperlink	-hyperlink	laymch	-laymch
laydel	-laydel	mledit	-mledit
laymrg	-laymrg	partiaload	-partiaload
parameters	-parameters	publish	-publish, +publish
pointcloudattach	-pointcloudattach	renderpresets	-renderpresets
purge	-purge	toolbar	-toolbar
table	-table	visualstyles	-visualstyles, vscurrent, vssave
view	-view		
wssave	-wssave		

Creating a Script

Since a script file is a plain ASCII text file, you can create and edit it with Notepad on Windows or TextEdit on Mac OS.

NOTE The examples in this section assume that you created a folder named `MyCustomFiles` in your `Documents` (or `My Documents`) folder on your local drive. If you have not created this

folder, do so now. If you created the folder in a different location, make sure you adjust the steps accordingly.

The following explains how to create a script file named `startup.scr` on Windows:

1. Do one of the following:
   - On Windows XP or Windows 7, click the Start button ➢ [All] Programs ➢ Accessories ➢ Notepad.
   - On Windows 8, on the Start Screen, type **note** and then click Notepad.
2. In Notepad, click File ➢ Save As.
3. In the Save As dialog box, browse to the `MyCustomFiles` folder that you created under the `Documents` (or `My Documents`) folder, or the location where you want to store the script file.
4. In the File Name text box, type **`startup.scr`**.
5. Click the Encoding drop-down list and select ANSI. Click Save.
6. In the editor, type a semicolon to start a comment line. Follow the semicolon with the current date and your name. If you want, feel free to add comments to explain what the script is used for. The following is an example of how your comment might look:

   ```
   ; Created 10/1/2013 by Lee Ambrosius - Startup Script
   ```

7. In the editor, type the following. Be sure to press Enter after each line, even after the last line of the script. Press the spacebar instead of typing {SPACE} near the end of the third line of the script.

   ```
   grid off
   ltscale 48
   -layer new border color 9 border set border{SPACE}
   rectang 0,0 1056,816
   -text 20,20 12 0 This is a sample startup script.
   zoom extents
   ```

8. Click File ➢ Save.

If you are running AutoCAD on Mac OS, use the following steps to create a script file named `startup.scr`:

1. In the Mac OS Finder, click Go ➢ Applications. In the Finder window, double-click TextEdit.
2. In TextEdit, click TextEdit ➢ Preferences. In the Preferences dialog box, on the New Document tab click Plain Text and then close the dialog box.
3. Click File ➢ New to create a plain ASCII text file.
4. Click File ➢ Save and type **`startup.scr`** in the Save As text box. From the sidebar on the left, click Documents ➢ MyCustomFiles, or the location you want to store the script file in, and then click Save.

5. In the editor, type a semicolon to start a comment line. Follow the semicolon with the current date and your name. If you want, feel free to add comments to explain what the script is used for. The following is an example of how your comment might look:

```
; Created 10/1/2013 by Lee Ambrosius - Startup Script
```

6. In the editor, type the following. Be sure to press Enter after each line, even after the last line of the script. Press the spacebar instead of typing {SPACE} near the end of the third line of the script.

```
grid off
ltscale 48
-layer new border color 9 border set border{SPACE}
rectang 0,0 1056,816
-text 20,20 12 0 This is a sample startup script.
zoom extents
```

7. Click File ➢ Save.

Running a Script

After you create a script file, you can run it using the `script` command. You can also run a script when AutoCAD first starts up using the `/b` and `-b` command-line switches. I covered command-line switches in Chapter 4, "Manipulating the Drawing Environment."

The following explains how to run a script file named `startup.scr` at the AutoCAD command prompt. You can create the script file by completing the example under the "Creating a Script" section earlier in this chapter, or you can obtain the file by downloading the book's sample files from `www.sybex.com/go/autocadcustomization`. Once downloaded, extract the files to a folder named `MyCustomFiles` under the `Documents` (or `My Documents`) folder of your user profile, or a folder of your choice.

1. Do one of the following:

   - On the ribbon, click Manage tab ➢ Applications panel ➢ Run Script (Windows).
   - At the command prompt, type **`script`** and press Enter. (You could also use the **`-script`** command to enter the location and name of the script you want to run at the command prompt instead of using the Select Script File dialog box—on Windows and Mac OS.)

2. In the Select Script File dialog box, browse to and select the `startup.scr` file. Click Open.

While a script is executing, you can press the Esc or Backspace key to pause it. After a script is paused, you can start running the script where it left off by issuing the `resume` command. Once a script is finished, you can run the script again using the `rscript` command. An additional command that is related to working with scripts is the `delay` command, which is used to pause a script for a specified duration of time that is expressed in milliseconds.

TIP On Windows only, you can run a script file across multiple drawings in a batch process using Script Pro, which can be downloaded for free from the Autodesk website:

`http://usa.autodesk.com/adsk/servlet/item?siteID=123112&id=4091678&linkID=9240618`

Recording Action Macros (Windows Only)

Action macros were first introduced with AutoCAD 2009 and are the next evolution of script files. Scripts and action macros share some similarities, but as I demonstrate in this section, action macros are easier to create because you create them interactively from the AutoCAD user interface instead of using Notepad.

An action macro is created using the Action Recorder on the ribbon, and then saved to a file with the `.actm` extension. Once an action macro is saved, you can edit the values in the recorded actions and the behavior of the action macro using the Action tree of the Action Recorder. Playing back an action macro is as simple as starting a standard AutoCAD command.

What Is an Action?

An *action* is the smallest user interaction or task that can be recorded from the AutoCAD user interface with the Action Recorder. An action might be the result of starting a command, entering a value at the command prompt, specifying a point in the drawing area, or even changing an object's property value with the Properties palette.

NOTE While you can use commands that display dialog boxes when recording an action macro, I recommended not doing this. The Action Recorder can't record the changes that are made within a dialog box and can produce unexpected results during the playback of an action macro. Instead, you should use commands that display options at the command prompt; see Table 8.1 and Table 8.2 under the section "Alternatives to Dialog Boxes" earlier in this chapter for more information.

While the command prompt will serve as your main method for starting commands and specifying command options and values, you can also use a variety of other methods for starting commands and providing input. The following lists most of the other ways that you can start a command or change a value using the user interface when recording an action macro:

- Application menu and Quick Access toolbar
- Ribbon panels, toolbars, and pull-down and shortcut menus
- Application and drawing-window status bars
- Properties and Quick Properties palettes
- Tool Palettes window
- Input provided with the input device (coordinate values and selection sets)

There are some other types of commands that you should avoid in addition to ones that display dialog boxes. These commands are related to opening, creating, and closing drawings, as they (along with a few other commands) can't be recorded with the Action Recorder. If you

attempt to enter a command that can't be used while the Action Recorder is in recording mode, the following message will be displayed in the Command Line History window:

```
** ACTRECORD command not allowed during recording an action macro **
```

Using the Action Recorder

The Action Recorder is a panel on the ribbon, as shown in Figure 8.1, which you use to record actions so they can be played back later to automate a repetitive task. You also use the Action Recorder panel to modify and manage previously recorded action macros. You can display the Action Recorder by clicking the Manage tab on the ribbon.

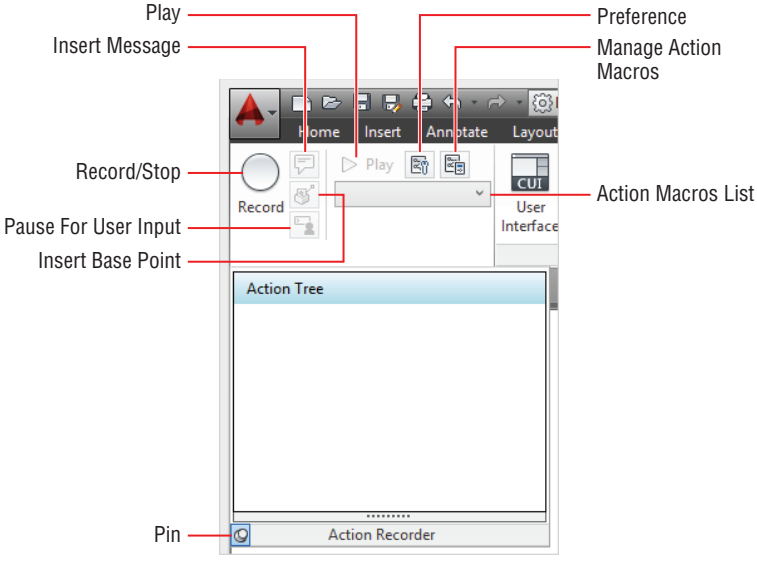


FIGURE 8.1 Action Recorder panel on the ribbon

The following explains the various components of the Action Recorder panel:

Record/Stop The Record button starts the recording of actions (`actrecord` command). After you are done recording actions, clicking the Stop button displays the Action Macros dialog box (`actstop` command). From the Action Macros dialog box, you can choose to save or discard the recorded actions. Saving the recorded actions results in the creation of an ACTM file.

Insert Message The Insert Message button allows you to add to the current action macro in the Action Tree an action that displays a message box with a custom message during playback. When clicked, the Insert User Message dialog box is displayed. You can also use the `actmessage` command when you are recording an action macro.

Play The Play button starts the playback of the action macro shown in the Action Macros drop-down list. An action macro can also be played back by entering its name (without the file extension) at the command prompt.

Preference The Preference button displays the Action Recorder Preferences dialog box, which allows you to control the settings that affect the Action Recorder panel when creating or playing back an action macro.

Manage Action Macros The Manage Action Macros button displays the Action Macro Manager, which allows you to copy, rename, modify, and delete the action macro files found in the locations specified on the Files tab of the Options dialog box. You can also use the `actmanager` command to display the Action Macro Manager.

Action Macros List The Action Macros drop-down list displays all of the action macros that are available for playback, beginning with the ones you have most recently played back. Recently played action macros are located at the top of the list. Below the separator bar, all available action macros are listed. Clicking the Manage Action Macros item at the bottom of the drop-down list displays the Action Macro Manager.

Action Tree The Action Tree displays all of the actions and values that make up a currently selected action macro or the action macro that is currently being recorded.

Insert Base Point The Insert Base Point button adds a base point-action to the current action macro in the Action Tree. A base point action establishes a new absolute coordinate point that the next relative point uses as its reference point. You can also use the `actbasepoint` command when you are recording an action macro.

Pause For User Input The Pause For User Input button allows you to prompt the user to specify a new value for an action during playback. This button acts like a toggle for the selected action value in the Action Tree; it switches a value from static to interactive or an interactive value back to a static value. You can also use the `actuserinput` command when you are recording an action macro.

Recording Actions for Playback

Recording actions in AutoCAD to create an action macro is similar to recording your favorite movie or TV show using a digital video recorder. You click the Record button (`actrecord` command) to start recording actions that are performed in the AutoCAD user interface. Once recording has started, the button changes from Record to Stop. After you are done recording actions, click the Stop button (`actstop` command) and then save or discard the recorded actions.

The following example explains how to record an action macro that creates a layer named Hardware and inserts a block from the Tool Palettes window.

1. Display the Tool Palettes window. (On the ribbon, click View tab ➢ Palettes panel ➢ Tool Palettes.) You want to display the Tool Palettes window before you start recording; otherwise the action that opens the window will be recorded as part of the action macro.

2. On the ribbon, click Manage tab ➢ Action Recorder panel ➢ Record.
3. At the command prompt, type **`-layer`** and press Enter. You should now see a new node in the Action Tree that shows it has recorded the start of the `-layer` command; see Figure 8.2.
4. At the `Enter an option [?/Make/Set/New/Rename/ON/OFF/Color/Ltype/LWeight /TRansparency/MATerial/Plot/Freeze/Thaw/LOck/Unlock/stAte/Description /rEconcile]:` prompt, type **`m`** and press Enter.

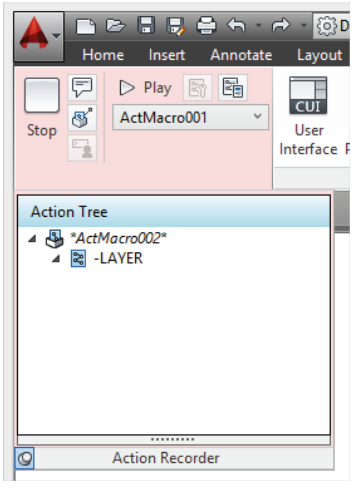


FIGURE 8.2
The Action Tree shows recently recorded actions.

5. At the `Enter name for new layer (becomes the current layer) <0>:` prompt, type **`Hardware`** and press Enter.
6. At the `Enter an option [?/Make/Set/New/Rename/ON/OFF/Color/Ltype/LWeight/TRansparency/MATerial/Plot/Freeze/Thaw/LOck/Unlock/stAte/Description/rEconcile]:` prompt, type **c** and press Enter.
7. At the `New color [Truecolor/COlorbook]:` prompt, type **5** and press Enter.
8. At the `Enter name list of layer(s) for color 5 (blue) <Hardware>:` prompt, press Enter. In the Action Macro - Value Not Recorded dialog box, click Use The Value That Is Current At Playback.
9. At the `Enter an option [?/Make/Set/New/Rename/ON/OFF/Color/Ltype/LWeight/TRansparency/MATerial/Plot/Freeze/Thaw/LOck/Unlock/stAte/Description/rEconcile]:` prompt, press Enter to end the `-layer` command. Figure 8.3 shows all of the values recorded by the Action Recorder for the `-layer` command.
10. In the Tool Palettes window, right-click the title bar and click Manufacturing.

11. On the Mechanical tab, click the Hex Nut - Metric tool.
12. At the `Specify insertion point or [Basepoint/Scale/X/Y/Z/Rotate]:` prompt, click in the drawing area.

13. On the ribbon, click Manage tab ➤ Action Recorder panel ➤ Stop.
14. When the Action Macro dialog box (see Figure 8.4) is opened, in the Action Macro Command Name text box type **HexNut**.
15. In the Description text box, type **Inserts the HexNut block on the Hardware layer**.
16. Optionally, in the Restore Pre-playback View section, choose the options you want to use during the playback of your action macro.

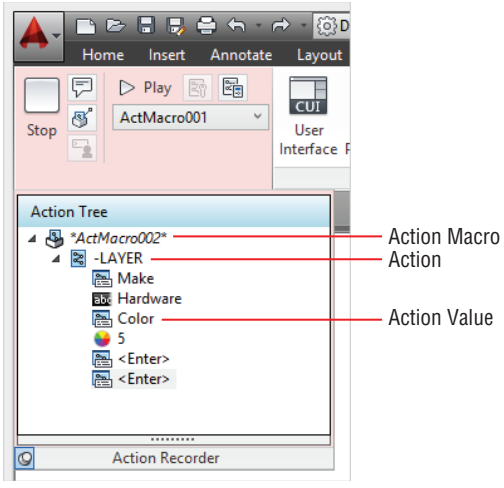


FIGURE 8.3
Recorded actions and values used to create a new layer named Hardware with the color blue

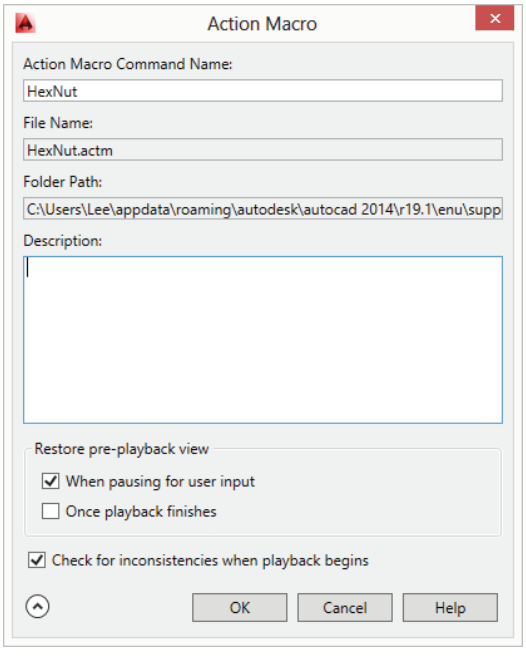


FIGURE 8.4
Saving recorded actions to an action macro

17. Optionally, check (or clear) Check For Inconsistencies When Playback Begins. When this option is selected, AutoCAD checks a set of predefined conditions in the current drawing against those that were used when the action macro was created. If differences are identified that could affect the playback of the action macro, a message box is displayed.

18. Click OK to create a file named `HexNut.actm`.

Now that you have created an action macro file, you can play it back by doing one of the following:

- At the command prompt, enter the name of the action macro you want to execute. (The name of the action macro is the same as the action macro's file, but without its file extension.)
- On the Action Recorder panel, select from the Action Macros list the action macro you want to play back, and click Play.
- Right-click in the drawing area, and then click Action Recorder ➢ Play and the action macro you want to execute from the submenu.

The following example explains how to execute the HexNut action macro you just created:

1. Create a new drawing. While you can play back an action macro in the drawing you used to record the action macro, you will not be able to see if the block is inserted or the layer is created.
2. At the command prompt, type **`hexnut`** and press Enter. Playback of the action macro, which creates the Hardware layer and then inserts the HexNut block at the same point specified by the action macro, is started.
3. In the Action Macro - Playback Complete message box, click Close. If the message box is not displayed, it is because someone previously checked the Do Not Show Me This Message Again option before they clicked Close.

TIP You can redisplay a message box that you previously clicked the Do Not Show option to dismiss. To redisplay the message box, display the Options dialog box (`options` command), and click the System tab. On the System tab, click Hidden Message Settings. In the Hidden Message Settings dialog box, select the message box you want to redisplay the next time it is called, and then click OK twice to save the changes.

The following explains how to change the action macro to prompt for an insertion point when the block is being inserted:

1. On the ribbon, click Manage tab ➢ Action Recorder panel, and then click the panel's title bar to expand it.

2. In the Action Tree, right-click the coordinate value below the EXECUTETOOL action node. Click Pause For User Input.
3. At the command prompt, type **`hexnut`** and press Enter.
4. At the `Specify insertion point or [Basepoint/Scale/X/Y/Z/Rotate]:` prompt, specify a coordinate value in the drawing window.

I will discuss other ways that you can modify an action macro in the next section.

NOTE You can find the completed action macro from this section as part of the sample files for this book at `www.sybex.com/go/autocadcustomization`. Once downloaded, place the `HexNut.actm` file in the folder that is stored in the `actrecpath` system variable.

Modifying Action Macros

After an action macro has been created and saved, you can modify its properties or the actions that make it up. You can perform the following tasks to modify an action macro:

- Pause an action macro for input
- Edit the value of an action
- Toggle a coordinate or all coordinates between relative and absolute
- Insert a user message into an action macro
- Insert a base point
- Remove an action from an action macro

Pausing for Input

You can have an action macro request a new value for a previously recorded value by doing the following:

1. On the ribbon, click Manage tab ➢ Action Recorder panel, and then click the panel's title bar to expand it.
2. In the Action Tree, select the action value node that you want the action macro to pause at and prompt for a new value.
3. Do one of the following:

   - In the main area of the Action Recorder panel, click Pause For User Input.
   - In the Action Tree, right-click a node and click Pause For User Input.

Editing a Recorded Value

A recoded value can be changed by following these steps:

1. On the ribbon, click Manage tab ➢ Action Recorder panel, and then click the panel's title bar to expand it.
2. In the Action Tree, right-click a value and click Edit.
3. In the in-place editor, type a new value and press Enter.

Toggling between Absolute and Relative Coordinate Values

Recorded coordinate values are all relative to the first coordinate point that is specified when recording actions. You can toggle a relative coordinate value to an absolute coordinate that you specified in the drawing window while the value was being recorded. Use the following example to toggle a coordinate value from relative to absolute or absolute to relative:

1. On the ribbon, click Manage tab ➢ Action Recorder panel, and then click the panel's title bar to expand it.

2. In the Action Tree, right-click a coordinate value and click Relative To Previous. When checked, the coordinate value is relative to the previous coordinate value in the action macro.

If you want to make all coordinates absolute or relative with the exception of the first coordinate value, do the following:

1. On the ribbon, click Manage tab ➢ Action Recorder panel, and then click the panel's title bar to expand it.
2. In the Action Tree, right-click the topmost node and click All Points Are Relative. When checked, the first coordinate value is absolute and all others are relative to the previous coordinate value.

Inserting a User Message

You can display a message box during the playback of an action macro by doing the following:

1. On the ribbon, click Manage tab ➢ Action Recorder panel, and then click the panel's title bar to expand it.
2. In the Action Tree, select a node where you would like to display a user message before the action is executed. If you select the topmost node, the user message is inserted as the first action of the action macro.
3. Do one of the following:

   - In the main area of the Action Recorder panel, click Insert Message.
   - In the Action Tree, right-click a node and click Insert User Message.
4. When the Insert User Message dialog box opens, type the message you want to display in the message box and then click OK.

Inserting a Base Point

A base point in an action macro allows you to establish an absolute coordinate point that the next relative coordinate point references. The following steps explain how to insert a base point into an action macro:

1. On the ribbon, click Manage tab ➢ Action Recorder panel, and then click the panel's title bar to expand it.
2. In the Action Tree, select the node that you want to insert a base point before. If you select the topmost node, the base point is inserted as the first action of the action macro.
3. Do one of the following:

   - In the main area of the Action Recorder panel, click Insert Base Point.
   - In the Action Tree, right-click a node and click Insert Base Point.
4. At the `Specify a base point:` prompt, specify a coordinate value in the drawing window.

Removing an Action

Actions, base points, and user messages can be removed from an action macro, but an individual recorded value can't be removed. For example, if you recorded the Color option of the `layer` command, you can't remove the Color value that was entered. However, you can remove the action that contains the recording of the `layer` command and any responses to the `layer` command. You can remove an action, base point, or user message by doing the following:

1. On the ribbon, click Manage tab ➢ Action Recorder panel, and then click the panel's title bar to expand it.
2. In the Action Tree, right-click an action, base point, or user message node and click Delete.
3. In the Action Macro - Confirm Deletion of Action Node message box, click Delete.

Managing Action Macros

After action macros have been saved to ACTM files, you can manage the files with the Action Macro Manager (`actmanager` command). The following tasks can be performed from the Action Macro Manager:

- Copy an action macro
- Rename an action macro
- Delete an action macro
- Edit the properties of an action macro

Creating a Copy of an Action Macro

You can create a copy of an existing action macro by doing the following:

1. On the ribbon, click Manage tab ➢ Action Recorder panel ➢ Manage Action Macros.
2. When the Action Macro Manager opens, in the Action Macros list, select the action macro you want to copy. Click Copy.
3. When the Action Macro dialog box is opened, in the Action Macro Command Name text box enter a new name.
4. Optionally, change the other settings in the dialog box as desired.
5. Click OK to finish copying the action macro.
6. Click Close to return to the drawing window.

TIP You can also copy an ACTM file using Windows Explorer or File Explorer. When copying a file in Windows Explorer or File Explorer, make sure the name of the new file does not contain spaces or special characters and contains fewer than 31 characters. If the name of the file is invalid, the action macro cannot be loaded into AutoCAD.

Renaming an Action Macro

You can rename an existing action macro by doing the following:

1. On the ribbon, click Manage tab ➢ Action Recorder panel ➢ Manage Action Macros.
2. When the Action Macro Manager opens, in the Action Macros list select the action macro you want to rename. Click Rename.
3. In the in-place editor, type the new name and press Enter.
4. Click Close to return to the drawing window.

Deleting an Action Macro

You can delete an existing action macro by doing the following:

1. On the ribbon, click Manage tab ➢ Action Recorder panel ➢ Manage Action Macros.
2. When the Action Macro Manager opens, in the Action Macros list select the action macro you want to remove. Click Delete.
3. In the Action Macro - Confirm Deletion message box, click Delete.

TIP An ACTM file also can be deleted in Windows Explorer or File Explorer.

Changing the Properties of an Action Macro

After an action macro has been saved, you can change its properties by doing the following:

1. On the ribbon, click Manage tab ➢ Action Recorder panel and click the panel's title bar to expand it.
2. In the Action Tree, right-click the topmost node and click Properties.
3. When the Action Macro dialog box opens, make the changes to the action macro and click OK.

Sharing and Specifying Paths for Action Macros

Action macro files can be shared with others by simply copying ACTM files from one workstation to another, or posting them to a shared location. While copying an ACTM file is straightforward, all workstations that the action macro will be played back on must have access to the following:

- Commands that were used during the recording of the action macro
- Files that were used by the commands in the action macro

AutoCAD loads any ACTM files it finds in the folders listed under the Action Recorder Settings node on the Files tab of the Options dialog box (`options` command). The Action Recorder Settings node contains two child nodes:

Actions Recording File Location The folder listed under this node is used to store newly recorded ACTM files. Files located in the folder can also be edited and played back. You can also use the `actrecpath` system variable to query and set the folder used by the Action Recorder.

Additional Actions Reading File Location The folders listed under this node are searched for additional ACTM files that can be played back only. This node is where you might specify a network location that contains all of the ACTM files that are shared with everyone in your company or department. You can also use the `actpath` system variable to query and set the folders used by the Action Recorder.

Chapter 9

Defining Shapes, Linetypes, and Hatch Patterns

The Autodesk® AutoCAD® program uses shapes, linetypes, and hatch patterns to control the way text, linework, and filled areas look. Shapes are the foundation for the letters of an SHX font file, but they can also be inserted into a drawing or used to help communicate design information with complex linetypes.

All objects in a drawing are affected by linetypes because they control how the line work of an object appears. You can create your own custom linetypes with dash, dot, and gap combinations different from those that come with AutoCAD and even include a text string or shape. You can also create custom hatch patterns to introduce new linework to fill a closed area.

Creating and Compiling Shapes

Shapes are objects that can represent straight and curved linework in a single object. Over the years, shapes have held several different roles in designs created with AutoCAD. Shapes were the original concept that eventually evolved into blocks, and they are still are used today to define the letters in an SHX font file and to display objects in a complex linetype.

Shapes are stored in files with the `.shp` extension. SHP files are in the ASCII text format and can be edited with Notepad (Windows) or TextEdit (Mac OS). A SHP file can contain multiple shapes. Each shape definition in a SHP file must have a unique name, and in the case of creating a font file, the names must follow a specific numbering convention as well. After you create a SHP file, it must be compiled into an SHX file (using the `compile` command) before any of the shapes in the file can be used with AutoCAD.

TIP On Windows, you can use the Express Tools Make Shape tool to define a shape without needing to know the syntax. Click Express tab ➢ Tools panel ➢ panel's title bar ➢ Make Shape.

Structure of a Shape

Each shape is composed of two lines: header and specification. The header (or first) line contains three components: the shape number, the number of bytes in the shape specification, and a name. The syntax of the header line for a shape is as follows:

```
*shape_number,definition_bytes,shape_name
```

Here is an example of a header line for a shape that has the number 136, contains 6 bytes, and is named DMOND:

```
*136,6,DMOND
```

When creating the header line for a shape, keep the following in mind:

- The line must begin with an asterisk.
- The line length cannot exceed 128 characters.

Table 9.1 provides more information about each of the components that make up the header line of a shape.

TABLE 9.1: Components of a shape header line

CODE	DESCRIPTION
`shape_number`	Unique number for a shape within the file. All shapes must have an assigned number between 1 and 258, but the number can be increased up to 32,768 for Unicode fonts. When you are defining a font, keep in mind that the numbers 256 (U+00B0), 257 (U+00B1), and 258 (U+2205) are reserved for the degree sign, plus or minus sign, or diameter symbol, respectively.
`definition_byte`	Total sum of bytes used in the shape specification line. Specification groupings contained in parentheses are not counted as a single byte, but each item in the grouping is counted as a single byte.
`shape_name`	Name used to identify the shape for insertion into a drawing or use as part of a complex linetype. The name must be specified in all uppercase letters.

The second and subsequent lines of the shape specification consist of a comma-delimited list of code values that define the objects that make up the appearance of the shape. Each shape specification must end with `0` (zero). The syntax of the specification line looks like this:

```
spec1,spec2,specN,...,0
```

Here is an example of a shape specification that draws four line segments at four different angles to form a diamond:

```
1,023,02D,02B,025,0
```

Table 9.2 explains the codes that can make up a specification line of a shape. Figure 9.1 shows the available vector or octant angles that can be used when drawing straight vectors or octant arcs.

TABLE 9.2: Shape specification line codes

CODE	DESCRIPTION
`000` or `0`	Represents the end-of-shape definition marker. All shape definitions must end with this code; otherwise an error is generated when the shape file is compiled. The error message displayed is `Premature end of file`. Syntax: `0`
`001` or `1`	Controls the *Pen Down* drawing mode. Used to draw straight vectors. Syntax: `1,0`*LA* *L*: `Vector length.` Valid range is 0 to 15 units. For values 10 to 15, use hex representation (letters A to F). *A*: Vector angle. Valid range is 0 to 15, where each number represents an increment of about 22.5 degrees. For values 10 to 15, use hex representation (letters A to F).
`002` or `2`	Controls the *Pen Up* drawing mode. Used to move the pen to a new location without drawing a vector. Syntax: `2,0`*LA* *L*: `Vector length.` Valid range is 0 to 15 units. For values 10 to 15, use hex representation (letters A to F). *A*: Vector angle. Valid range is 0 to 15; each number represents an increment of about 22.5 degrees. For values 10 to 15, use hex representation (letters A to F).
`003` or `3`	Scales down by a specified factor any vectors that follow. Syntax: `3,`*Div* *Div*: `Divisor or scale reduction factor.` Valid range is 0 to 15 units. For values 10 to 15, use hex representation (letters A to F).
`004` or `4`	Scales up by a specified factor any vectors that follow. Syntax: `4,`*Mul* *Mul*: `Multiplier or scale factor.` Valid range is 0 to 15 units. For values 10 to 15, use hex representation (letters A to F).
`005` or `5`	Saves, or pushes, the current position in the shape. Doing so allows you to return to the position and continue drawing. A maximum of four locations can be saved at any one time. If the maximum number of saves is exceeded, the error message `Position stack overflow in shape <number>` is displayed. Syntax: `5`

TABLE 9.2: Shape specification line codes *(CONTINUED)*

CODE	DESCRIPTION
006 or 6	Restores, or pops, the most recently saved position. A maximum of four positions can be restored. The positions are returned from most recent to earliest saved. If you attempt to restore more positions than those stored, the error message `Position stack underflow in shape <number>` is displayed. Autodesk recommends that you restore all saved positions before you add the end-of-shape definition marker. Not restoring all saved positions could affect other shapes. Syntax: 6
007 or 7	References another shape defined in the same shape file. By referencing other shapes, it allows you to define new shapes without having to duplicate or create long shape definitions. Syntax: 7,*Num* *Num*: Number of the shape to use in the current file.
008 or 8	Draws a vector to a given coordinate value, thereby giving you much greater control over the vectors that can be drawn when compared to what can be done using the codes 0 (*Pen Down*) and 1 (*Pen Up*). Coordinate value range is –128 to 127 along the X- or Y-axis. Optionally, you can put the coordinates in parentheses to make a shape definition easier to read. Syntax: 8,*X*,*Y* or 8,(*X*,*Y*) *X*: Displacement along the X-axis. *Y*: Displacement along the Y-axis.
009 or 9	Draws multiple vectors based on coordinate pairs. Coordinate value range is –128 to 127 along the X- or Y-axis. Optionally, you can put the coordinates in parentheses to make a shape definition easier to read. Syntax: 9,X_1,Y_1,X_2,Y_2,X_n,Y_n or 9,(X_1,Y_1),(X_2,Y_2),(X_n,Y_n) *Xn*: Displacement along the X-axis. *Yn*: Displacement along the Y-axis.
00A or 10	Draws an arc; more specifically, it draws an *octant arc* because the arc can span one or more 45-degree octants. The arc must start and end at one of the octants. Each octant is numbered counterclockwise from 0 to 7. Figure 9.1 shows each octant value and the equivalent angle in degrees. Syntax: 10,*Rad*,0*SE* *Rad*: Radius of the arc expressed in radians. Valid range is 1 to 255. *S*: Start octant for the arc. *E*: End octant for the arc.

TABLE 9.2: Shape specification line codes *(CONTINUED)*

CODE	DESCRIPTION
00B or 11	Draws an arc that doesn't follow the octant boundaries like those drawn with code 10. This type of arc is known as a *fractional arc*. Although using code 11 provides greater flexibility when creating an arc, it requires additional planning and calculating to get the expected results. The start of the arc is calculated by taking the difference in degrees between the starting octant and the start of the arc. After the difference is calculated, you multiply the number by 256 and then divide the new value by 45. This gives you the start offset value. The end of the arc is calculated by taking the difference between the last octant boundary crossed and the end of the arc. After the difference is calculated, you multiply the number by 256 and then divide the new value by 45. This gives you the end offset value. Syntax: 11,*SO*,*EO*,*HRad*,*Rad*,*0SE* *SO*: Start offset from octant for the arc (0 when start offset is on an octant). *EO*: End offset from octant for the arc (0 when end offset is on an octant). *HRad*: High radius for the arc, the value is 0 (zero) if *Rad* is less than 255. To generate an arc greater than 255, multiply the *HRad* value by 256 and add it to Rad. *Rad*: Radius of the arc expressed in radians. Valid range is 1 to 255 when *HRad* is set to 0 (zero). *S*: Start octant for the arc. *E*: End octant for the arc.
00C or 12	Draws an arc based on a displacement on the XY plane and a bulge value. This type of arc is known as a *bulge fractional arc*. Coordinate value range is –127 to 127 along the X- or Y-axis, and the bulge value range is –127 to 127. Syntax: 12,*X*,*Y*,*B* *X*: Displacement along the X-axis. *Y*: Displacement along the Y-axis. *B*: Bulge of the arc. The bulge is calculated using the formula ((2 × *H*/*D*) × 127), where *H* is the height of the bulge (measured from the midpoint of the arc's chord to the arc's circumference). *D* is the length of the arc's chord, which is the distance between the start and endpoint of the arc.

TABLE 9.2: Shape specification line codes *(CONTINUED)*

CODE	DESCRIPTION
00D or 13	Draws multiple arcs based on coordinate pairs and bulge values, which are the same as code 00C. Coordinate value range is –127 to 127 along the X- or Y-axis, and the bulge value range is –127 to 127. Optionally, you can put the values in parentheses to make the shape definition easier to read. You must add 0,0 after the last bulge. Syntax: 13,X_1,Y_1,B_1,X_2,Y_2,B_2,X_n,Y_n,B_n,0,0 or 13,(X_1,Y_1,B_1),(X_2,Y_2,B_2),(X_n,Y_n,B_n),(0,0) *X*: Displacement along the X-axis. *Y*: Displacement along the Y-axis. *B*: Bulge of the arc. The bulge is calculated using the formula ((2 ×H/D) ×127). H is the height of the bulge, which is measured from the midpoint of the arc's chord to the arc's circumference. D is the length of the arc's chord, which is the distance between the start and endpoint of the arc.
00E or 14	Defines that the shape can be used with a vertical orientation when defining a font. When creating text with a vertical orientation, the specifications following the code are used. Commonly, this code is used to relocate the start and endpoints of the letter. I discuss creating a font later in this chapter in the "Defining a Font" section. Syntax: 14

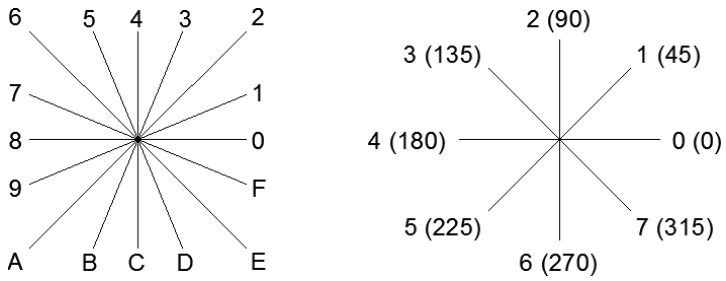


FIGURE 9.1 Determining which value to use for vector angles (left) or octants (right)

NOTE The shape_samples.shp file that is part of the sample files for this book contains a variety of shape definitions that demonstrate the use of the different codes. The sample files can be downloaded from www.sybex.com/go/autocadcustomization.

Now that you have an understanding of the syntax for a shape definition, here is what the definition of the standard BAT shape looks like from the ltypeshp.shx file. The BAT shape is used with the Batting linetype (see Figure 9.2) defined in the acad.lin and acadiso.lin files.

```
*134,6,BAT
025,10,2,-044,02B,0
```

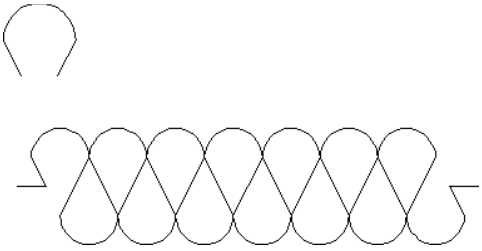


Figure 9.2
BAT shape inserted into the drawing (top) along with a polyline with the Batting linetype assigned (bottom)

NOTE The `ltypeshp.shx` file is a compiled file and can't be viewed using Notepad (Windows) or TextEdit (Mac OS). However, Autodesk does provide the original source for this file. The `ltypeshp.shp` file in the `Support` folder within the AutoCAD installation folder contains the source for the shapes in the `ltypeshp.shx` file.

Table 9.3 explains each code and value of the BAT shape definition.

Table 9.3: Codes and values for a BAT shape

Value	Description
`*`	Indicates the start of the shape
`134,`	Number assigned to the shape
`6,`	Length of the shape definition in bytes
`BAT`	Name of the shape
`025,`	Draw a vector from the current location approximately 2 units in the 5 direction (112.5 degrees)
`10,`	Indicates the start of an octant arc
`2,`	The radius of the octant arc
`-044,`	Start the arc in octant 4 (180 degrees) and go clockwise 4 octants (0 degrees)
`02B,`	Draw a vector from the current location approximately 2 units in the B direction (247.5 degrees)
`0`	End marker for the shape

Figure 9.3 shows a visual breakdown of how each part of the shape looks as AutoCAD interprets the BAT shape definition.

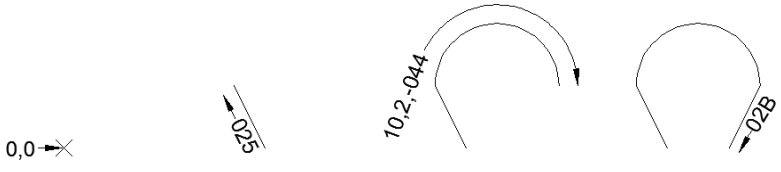


Figure 9.3
Breakdown of the BAT shape

Defining a Shape

Shape definitions are created and stored in SHP files using an ASCII text editor, such as Notepad (Windows) and TextEdit (Mac OS). While you could add your shape definitions to the `ltypeshp.shp` file and then recompile that file, I recommend storing your custom shape definitions in your own SHP files.

The following explains how to create a new SHP file named `myshapes.shp` on Windows:

1. Do one of the following:
   - Click Start button ➢ [All] Programs ➢ Accessories ➢ Notepad (Windows XP and Windows 7).
   - On the Start screen, type **note** and click Notepad (Windows 8).
2. In Notepad, click File ➢ Save As.
3. In the Save As dialog box, browse to the `MyCustomFiles` folder you created for the book in the `Documents` (or `My Documents`) folder, or to the location in which you want to create the SHP file.
4. In the File Name text box, type **`myshapes.shp`**.
5. From the Save As Type drop-down list, select All Files (*.*).
6. From the Encoding drop-down list, select ANSI. Click Save.
7. In the editor area, type the following:

   ```
   *136,6,DMOND

   1,023,02D,02B,025,0
   ```

8. Press Enter after the last line in the shape definition.
9. Click File ➢ Save.

If you are using AutoCAD on Mac OS, you can use these steps to create a new SHP file named `myshapes.shp`:

1. In the Mac OS Finder, click Go ➢ Applications.
2. In the Finder window, double-click TextEdit.
3. On the Mac OS menu bar, click TextEdit ➢ Preferences.
4. In the Preferences dialog box, click the New Document tab.
5. In the Format section, click Plain Text.
6. Close the Preferences dialog box.
7. Click File ➢ Save.
8. In the Save dialog box, type **`myshapes.shp`** in the Save As text box.
9. Browse to the `MyCustomFiles` folder you created for the book in the `Documents` folder, or to the location in which you want to create the SHP file.
10. From the Plain Text Formatting drop-down list, select Unicode (UTF-8).
11. Enable the If No Extension Is Provided, Use *.TXT check box, and then click Save.

12. In the editor area, type the following:

```
*136,6,DMOND
1,023,02D,02B,025,0
```

13. Press Enter after the last line in the shape definition.
14. Click File ➢ Save.

Figure 9.4 shows what the DMOND shape looks like when used in a drawing and also how the shape definition works.

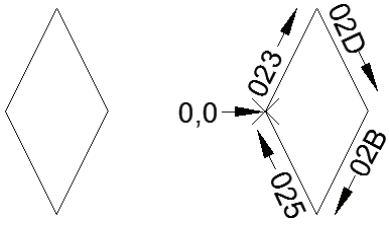


FIGURE 9.4
Inserted DMOND shape (left) and a breakdown of its shape definition (right)

Compiling a Shape

Before you can use a shape definition in a SHP file in AutoCAD, you must compile the file using the `compile` command. Compiling a SHP file validates the shape definitions in the file, and if there are no errors, the process generates an SHX file. The following explains how to compile the `myshapes.shp` file that I explained how to create in the "Defining a Shape" section.

NOTE You can also find the `myshapes.shp` file as part of the sample files for this book. The sample files can be downloaded from `www.sybex.com/go/autocadcustomization`.

1. At the Command prompt, type **`compile`** and press Enter.
2. In the Select Shape Or Font File dialog box, browse to the `MyCustomFiles` folder that you created for this book in the `Documents` (or `My Documents`) folder, or browse to the folder that contains the `myshapes.shp` file that you created if you placed it elsewhere.
3. Click Open. The following message is displayed if the command succeeds in compiling the file:

```
Compiling shape/font description file Compilation successful. Output file
myshapes.shx contains 49 bytes.
```

If an error message is displayed, go back to the SHP file and fix the shape definition.

Loading and Inserting a Shape

After a SHP file has been compiled as an SHX file, you can load the shapes into a drawing so they can be inserted using the `shape` command, or you can use the shape in a complex linetype. I cover creating complex linetypes later in this chapter in the "Creating and Using Custom Linetypes" section.

Before you proceed, make sure you have created and compiled the `myshapes.shp` file. The creation and compiling of the SHP file was discussed in the "Defining a Shape" and "Compiling a Shape" sections earlier in this chapter.

NOTE You can also find the `myshapes.shx` file as part of the sample files for this book. The sample files can be downloaded from `www.sybex.com/go/autocadcustomization`.

The following explains how to load the compiled shape (SHX) file named `myshapes.shx` into a drawing:

1. At the Command prompt, type **`load`** and press Enter.
2. In the Select Shape File dialog box, browse to the `MyCustomFiles` folder that you created for this book in the `Documents` (or `My Documents`) folder, or browse to the folder that contains the `myshapes.shx` file if you placed it elsewhere.
3. Click Open.

You can use the following steps to insert the shape named DMOND that was made available after loading the `myshapes.shx` file into the drawing file:

1. At the Command prompt, type **`shape`** and press Enter.
2. At the `Enter shape name or [?]:` prompt, type **`dmond`** and press Enter.
3. At the `Specify insertion point:` prompt, specify a coordinate point in the drawing.
4. At the `Specify height <1.0000>:` prompt, specify a numeric value and press Enter.
5. At the `Specify rotation angle <0>:` prompt, specify a numeric value and press Enter.

NOTE If you change the definition of a shape, you must remove all instances that use the shape definition from any drawing that you want to use the new shape definition in, and then remove the shape definition from those drawings with the `purge` command. After the shape definition has been removed, you can load the updated shape definition and then reinsert each of the shape objects.

Defining a Font

Using shapes, you can define your own letters to create a custom font. Much of what you need to know has already been explained earlier in this chapter. However, there are a few conventions that you need to follow when specifying a number or name for a shape that is used to define a font. Here are the requirements that you must follow when defining a shape font file:

- The value of the *`shape_number`* variable must correspond to the ASCII code value of the character it represents. To know what ASCII code value represents a specific character, you can use the AutoLISP® function `ascii`. For example, `(ascii "a")` returns an ASCII code value of 97, whereas `(ascii "A")` returns an ASCII code value of 65. As you can see from the examples, uppercase and lowercase letters return different values.
- The *`shape_name`* is commonly used to communicate whether the shape represents an uppercase or lowercase letter. However, unlike the shape numbers in a shape font file, the name doesn't hold any meaningful significance. The sample font files that are included in the AutoCAD Help system do follow a specific naming convention for uppercase and lowercase letters, though. For example, the sample file uses `uca` for the shape name of the uppercase letter `A` and `lca` for the lowercase `a` shape definition.

- A custom shape font file must define a line-feed shape, which is represented by the ASCII code value 10. The line feed does not draw a character but acts as a carriage return and drops down one line.
- *Pen Up* movements must be used to set the start and endpoint of a letter and to define the spacing between letters.
- Each shape font file must include a special header that defines the spacing above and below the baseline of uppercase and lowercase letters. The header must use the following syntax:

```
*UNIFONT,4,font_name
above,below,modes,0
```

For more information on creating a shape font file, see the AutoCAD Help system.

TIP AutoCAD includes two sample shape fonts that you can experiment with. The samples are included in the AutoCAD Help system and can be found by searching on the keywords "font sample." Copy the sample shape definitions to a SHP file and then compile the file so you can use it in AutoCAD.

Creating and Using Custom Linetypes

Linetypes are used to communicate the intent of the linework that makes up a design. Most of the linetypes that come with AutoCAD are universal to all types of designs, not just architectural or mechanical. But there are times when you want to create your own linetypes. You might want to create a new linetype in these situations:

- When a standard linetype doesn't display well in your drawings using the global linetype scale factor that other linetypes do. In this case, you might simply be creating a new linetype based on an existing linetype, but it might have longer dashes or wider gaps.
- When a standard linetype doesn't fit your needs. For example, while working on civil-engineering drawings, you might want to introduce new linetypes that help to communicate the location of a utility line or roadway offset.

AutoCAD supports two kinds of linetype:

- Simple linetypes are made up of gaps, dashes, and/or dots.
- Complex linetypes are made up of gaps, dashes, and/or dots, plus a text string or shape.

Linetypes are stored in files with the `.lin` extension. LIN files are in the ASCII text format and can be edited with Notepad (Windows) or TextEdit (Mac OS). The linetypes that AutoCAD comes with are stored in the `acad.lin` and `acadiso.lin` files. New linetypes can be added to these two files, or you can add them to a LIN file that contains the linetypes that have been approved for use as part of your company's standards.

I recommend storing the linetypes you create in a LIN file separate from those that come with AutoCAD. This makes it easier to back up your custom files and migrate to a future release. However, if you want to add your linetypes to one of the standard LIN files that come with

AutoCAD, place them after the `User Defined Linetypes` section near the bottom of the file. This section is denoted by the following text:

```
;;  User Defined Linetypes
;;
;;  Add any linetypes that you define to this section of
;;  the file to ensure that they migrate properly when
;;  upgrading to a future AutoCAD version.  If duplicate
;;  linetype definitions are found in this file, items
;;  in the User Defined Linetypes section take precedence
;;  over definitions that appear earlier in the file.
;;
```

TIP On Windows, you can use the Express Tools Make Linetype tool to create a linetype without needing to know the syntax. Click Express tab ➢ Tools panel ➢ panel's title bar ➢ Make Linetype.

Structure of a Linetype

Each linetype definition is made up of two lines. The first line defines the name and an optional description for the linetype:

```
*linetype_name[, description]
```

Here is an example of a linetype definition with the name DASH_DOT_DASH and a description:

```
*DASH_DOT_DASH, DDD __ . __ . __ . __ . __ . __ . __ . __
```

When naming or describing a linetype, keep the following in mind:

- The name must be preceded by an asterisk. This indicates the start of a new linetype definition.
- The name should be short and indicate the intended use. I recommend limiting a linetype name to 45 characters, as long names are harder to view in the AutoCAD user interface.
- Descriptions are optional, but I recommend adding them to your linetypes. If you do add a description, make sure to add a comma after the name of the linetype.
- The first line of a linetype definition is limited to 280 characters. This means that the linetype name, comma, and description combined must be 279 or fewer characters in total, because the asterisk counts as one character.

The second line of a linetype definition contains the actual values that describe how the pattern of the linetype should look:

```
A,val1,val2,valN,...
```

Here is an example of a linetype pattern that contains a dash, a gap, a dot, and a second gap:

```
A,.75,-.25,0,-.25
```

Each linetype pattern must start with an A, which indicates that pattern alignment is being used. Pattern alignment ensures that dashes are used as the endpoints of a linetype object instead of a gap or dot. After the pattern alignment indicator, a linetype pattern can include one or more numeric values that need to be separated by a comma.

The following describes the possible numeric values that can be used to define the pattern of a simple linetype:

- A positive number represents a dash (or line) segment.
- A zero represents a dot.
- A negative number represents a gap (or space).

Now that you understand the basics of a linetype definition, here is the definition of the standard Center linetype from the acad.lin file:

```
*CENTER,Center ____ _ ____ _ ____ _ ____ _ ____ _ ____
A,1.25,-.25,.25,-.25
```

Table 9.4 explains each of the values used to define the Center linetype.

TABLE 9.4: Values of the Center linetype

VALUE	DESCRIPTION
*	Indicates the start of the linetype
CENTER	Linetype name, a comma separator, and the linetype description
, Center ____ _ ____ _ ____ _ ____ _ ____ _ ____	Linetype description
A,	Pattern alignment flag
1.25,	Dash of 1.25 units when the LTSCALE system variable is set to 1
-.25,	Gap of 0.25 units
.25,	Dash of 0.25 units
-.25	Gap of 0.25 units

In addition to dashes, dots, and gaps, a linetype pattern can include a value that defines a text string or shape object. A text string or shape object value in a linetype pattern is designated

by a grouping of values within a set of square brackets. The syntax for a linetype pattern that includes a text string or shape object value is as follows:

```
["text_string",style_name,scale,rotation,x-offset,y-offset]

[shape_name,shape_filename,scale,rotation,x-offset,y-offset]
```

Table 9.5 explains each of the values used to add a text string and shape object to a linetype pattern.

TABLE 9.5: Text string and shape object values

VALUE	DESCRIPTION
`"text_string"`	Text to place in the linetype pattern.
`style_name`	Text style name to use for the text. The style name must exist in the drawing file for it to be used.
`shape_name`	Name of the shape to place in the linetype pattern.
`shape_filename`	Filename of the compiled shape (SHX) file that contains the shape being placed in the linetype. This file must be located in the AutoCAD support file search paths.
`scale`	Scale factor to be applied to the text or shape. Syntax: `S=#.#`
`rotation`	Rotation type and angle to be applied to the text or shape. You can choose from three rotation types: Upright (U)—Rotates the text or shape so it is displayed upright or near upright, and not upside down or backward. Relative (R)—Rotates the text or shape relative to the angle of the objects to which the linetype is applied. Absolute (A)—Rotates the text or shape so it always has the same orientation based on the world coordinate system, no matter the angle of the object to which the linetype is applied. Syntax: `U=#.#`, `R=#.#`, or `A=#.#`
`x-offset`	Shifts the text or shape along the X-axis on the linetype; a positive number moves the text or shape to the endpoint of the object to which the linetype is applied, whereas a negative number moves the text or shape to the start point. Syntax: `X=#.#`
`y-offset`	Shifts the text or shape along the Y-axis on the linetype; moves the text or shape up or down based on the direction of the object to which the linetype is applied. A positive value moves the text or shape up when an object is drawn left to right. Syntax: `Y=#.#`

Here are two of the standard linetype definitions from the `acad.lin` file. The first contains a text string and the second contains a shape object.

```
*HOT_WATER_SUPPLY,Hot water supply ---- HW ---- HW ---- HW ----
A,.5,-.2,["HW",STANDARD,S=.1,U=0.0,X=-0.1,Y=-.05],-.2

*FENCELINE2,Fenceline square ----[]-----[]----[]-----[]----[]---
A,.25,-.1,[BOX,ltypeshp.shx,x=-.1,s=.1],-.1,1
```

Defining a Custom Linetype

Linetype definitions are created and stored in LIN files using an ASCII text editor, such as Notepad (Windows) and TextEdit (Mac OS). As I previously mentioned, I recommend storing your custom linetypes in a LIN file that you create and not one that came with AutoCAD.

The following explains how to create a new LIN file named `mylinetypes.lin` on Windows:

1. Do one of the following:
   - Click Start button ➢ [All] Programs ➢ Accessories ➢ Notepad (Windows XP and Windows 7).
   - On the Start screen, type **note** and click Notepad (Windows 8).
2. In Notepad, click File ➢ Save As.
3. In the Save As dialog box, browse to the `MyCustomFiles` folder you created for the book in the `Documents` (or `My Documents`) folder, or to the location in which you want to create the LIN file.
4. In the File Name text box, type **`mylinetypes.lin`**.
5. From the Save As Type drop-down list, select All Files (*.*).
6. From the Encoding drop-down list, select ANSI. Click Save.

If you are using AutoCAD on Mac OS, you can use these steps to create a new LIN file named `mylinetypes.lin`:

1. In the Mac OS Finder, click Go ➢ Applications.
2. In the Finder window, double-click TextEdit.
3. On the Mac OS menu bar, click TextEdit ➢ Preferences.
4. In the Preferences dialog box, click the New Document tab.
5. In the Format section, click Plain Text.
6. Close the Preferences dialog box.
7. Click File ➢ Save.
8. In the Save dialog box, type **`mylinetypes.lin`** in the Save As text box.
9. Browse to the `MyCustomFiles` folder you created for the book in the `Documents` folder, or to the location in which you want to create the LIN file.

10. From the Plain Text Formatting drop-down list, select Unicode (UTF-8).
11. Enable the If No Extension Is Provided, Use *.TXT check box, and then click Save.

Simple Linetype

A simple linetype named DASH_DOT_DASH can be added to a LIN file by doing the following:

1. Start Notepad (Windows) or TextEdit (Mac OS), if the application is not already open.
2. Click File ➢ Open.
3. In the Open dialog box, browse to and select the `mylinetypes.lin` file that you created earlier in this section or the LIN file you want to edit. Click Open. On Windows, you will need to select All Files (*.*) from the Files Of Type drop-down list to see the LIN file.
4. In the editor window, type the following:

```
;; My Simple Linetypes
*DASH_DOT_DASH, DDD __ . __ . __ . __ . __ . __ . __ . __
A,.75,-.25,0,-.25
```

5. Click Save.

Figure 9.5 shows what the Dash Dot Dash linetype looks like when used in a drawing.

Figure 9.5
Dash Dot Dash linetype assigned to a polyline

Complex Linetype

The next example defines a complex linetype named PLOT_LINE that displays the text string PL:

1. Start Notepad (Windows) or TextEdit (Mac OS), if the application is not already open.
2. Click File ➢ Open.
3. In the Open dialog box, browse to and select the `mylinetypes.lin` file that you created earlier in this section or any other LIN file you want to edit. Click Open. On Windows, you will need to select All Files (*.*) from the Files Of Type drop-down list to see LIN files.
4. In the editor window, type the following:

```
;; My Complex Linetypes
*PLOT_LINE,Plot line ----PL----PL----PL----PL----PL----PL----
A,.25,-.125,["PL",STANDARD,S=.1,U=0.0,X=-0.1,Y=-.05],-.1
```

5. Click Save.

Figure 9.6 shows what the Plot Line linetype will look like when used in a drawing.

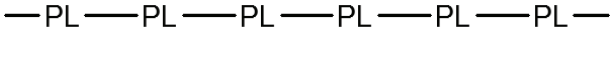


FIGURE 9.6
Plot Line linetype assigned to a polyline

Loading a Custom Linetype

Linetypes are loaded into a drawing using the Linetype Manager, which is displayed with the `linetype` command. The following explains how to load and use the custom linetypes Dash Dot Dash and Plot Line that you added to a LIN file in the "Defining a Custom Linetype" section. I explain how to adjust the scale factors that affect the display of linetypes in the "Controlling the Display of Linetypes" section.

NOTE You can find the custom linetypes in the file `mylinetypes.lin` that is part of the sample files for this book. The sample files can be downloaded from `www.sybex.com/go/autocadcustomization`.

For AutoCAD on Windows, use these steps:

1. On the ribbon, click Home tab ➢ Properties panel ➢ Linetypes drop-down list ➢ Other (or at the Command prompt, type **`linetype`** and press Enter).
2. When the Linetype Manager opens, click Load.
3. When the Load Or Reload Linetypes dialog box opens, click File.
4. In the Select Linetype File dialog box, browse to and select the `mylinetypes.lin` file that you created earlier in this section or the LIN file in which you added the custom linetypes. Click Open.
5. In the Available Linetypes list, press and hold the Ctrl key and then select the DASH_DOT_DASH and PLOT_LINE linetypes. Click OK.
6. In the Linetype Manager, select DASH_DOT_DASH and click Current.
7. Click OK to return to the drawing window.
8. Start the `pline` command and draw a new polyline with multiple segments.
9. On the ribbon, click Home tab ➢ Properties panel ➢ Linetypes drop-down list ➢ PLOT_LINE.
10. Start the `pline` command again and draw a new polyline with multiple segments.
11. At the Command prompt, type **`ltscale`** and press Enter.
12. At the `Enter new linetype scale factor <1.0000>:` prompt, type **`6`** and press Enter. You might need to adjust the global linetype scale factor or the length of the polyline segments to see the dashes, gaps, dots, and text displayed as part of the linetypes.

For AutoCAD on Mac OS, use these steps:

1. On the menu bar, click Format ➢ Linetype (or at the Command prompt, type **linetype** and press Enter).
2. When the Linetype Manager opens, click the + (plus sign).
3. When the Load Or Reload Linetypes dialog box opens, click the Files button located to the right of the Files drop-down list.
4. In the Select Linetype File dialog box, browse to and select the `mylinetypes.lin` file that you created earlier in this section or the LIN file in which you added the custom linetypes. Click Open.
5. In the Available Linetypes list, hold down the Command key and select the DASH_DOT_DASH and PLOT_LINE linetypes. Click Add.
6. In the Linetype Manager, double-click the DASH_DOT_DASH linetype.
7. Click Save to return to the drawing window.
8. Start the `pline` command and draw a new polyline with multiple segments.
9. On the Properties palette, click the Linetype drop-down list and select PLOT_LINE.
10. Start the `pline` command and draw a new polyline with multiple segments.
11. At the Command prompt, type **ltscale** and press Enter.
12. At the `Enter new linetype scale factor <1.0000>:` prompt, type **6** and press Enter. You might need to adjust the global linetype scale factor or the length of the polyline segments to see the dashes, gaps, dots, and text displayed as part of the linetypes.

TIP You can use the `load` option of the `-linetype` command on both Windows and Mac OS to load a custom linetype with a script, command macro, or AutoLISP program.

Controlling the Display of Linetypes

Four system variables control the scale factor that is used to adjust the size at which the dashes, gaps, text, and shapes that make up a linetype pattern appear. These system variables are as follows:

CELTSCALE Linetype scale factor that is applied to each new object created.

LTSCALE Linetype scale factor that is applied to all objects in a drawing.

MSLTSCALE Enables the scaling of the linetypes assigned to objects on the Model tab based on the current annotation scale. Use a value of 0 (zero) to disable the scaling of linetypes based on the current annotation scale; a value of 1 enables the scaling of linetypes. Default value is 1.

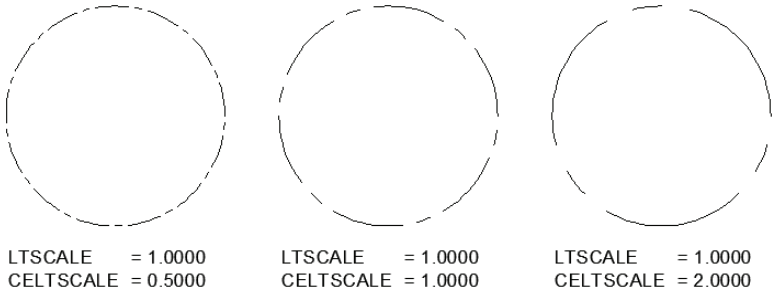
PSLTSCALE Enables the scaling of linetypes of objects displayed in the paper space viewports on a named layout. Use a value of 0 (zero) to scale the linetype for objects based on the

drawing units of the space they are drawn in and scaled by the global linetype scale factor stored in the `ltscale` system variable. Using a value of 1 scales all linetypes in model space or paper space equally based on the current scale factor stored in the `ltscale` system variable, no matter the current scale of each viewport.

Based on the values of the system variables I previously mentioned, AutoCAD calculates the linetype scale factor for each individual object in a drawing. The primary method used to calculate the scale factor that is applied to a linetype is to multiply an object's assigned linetype scale factor by the global linetype scale factor stored in the `ltscale` system variable.

For example, if an object is assigned a linetype scale factor of 0.75 and the `ltscale` system variable is set to 48, the final scale factor used to control the display of the linetype pattern applied to the object would be 36. If the object's individual linetype scale factor were 2 and `ltscale` were still 48, the final scale factor used to control the display of the linetype pattern applied to the object would be 96. You can see the results of assigning the individual linetype scale factors in Figure 9.7.

FIGURE 9.7
Same object with different linetype scale factors



Normally, most objects are assigned an individual linetype scale factor of 1. In situations where an object might be too small for the current global linetype scale, I recommend changing that object's linetype scale factor so the gaps, text, and shapes of the assigned linetype can be seen well. If you don't, the linetype pattern can appear as a single continuous dash.

Creating and Applying Custom Hatch Patterns

Hatch patterns allow you to fill closed areas with a repeating set of linework; they are commonly used on elevation and section views to help communicate the types of materials used. The hatch patterns that ship with AutoCAD represent a variety of materials that can be used in architectural, civil, or mechanical drawings.

However, the selection of hatch patterns that are included with AutoCAD are somewhat limited compared to all the materials that can be found on a building or manufactured product in the real world. For those materials that don't have a hatch pattern, you could create your own. Similar to linetype patterns, hatch patterns are defined with a series of dashes, gaps, and dots. Hatch patterns do support a few additional characteristics that allow you to control the offsets and angles of the linework that make up the hatch pattern.

Hatch patterns are stored in files with the `.pat` extension. PAT files are in the ASCII text format and can be edited with Notepad (Windows) or TextEdit (Mac OS). The hatch patterns that ship with AutoCAD are stored in the `acad.pat` and `acadiso.pat` files. New hatch patterns can be added to these files, or you can create your own PAT files. If you create your own PAT files, each hatch pattern must be in its own file and the name of the PAT file must be the same as the hatch-pattern name. For example, if you create a hatch pattern named pavement, the PAT filename must be `pavement.pat`.

I recommend storing the hatch patterns that you create in their own PAT files to keep them separate from those that come with AutoCAD. This makes it easier to back up your custom files and migrate to future releases. However, if you want to add your hatch patterns to one of the standard PAT files that come with AutoCAD, place them after the `User Defined Hatch Patterns` section near the bottom of the file. This section is denoted by the following text:

```
;;
;;  User Defined Hatch Patterns
;;  Add any hatch patterns that you define to this section of
;;  the file to ensure that they migrate properly when
;;  upgrading to a future AutoCAD version.  If duplicate hatch
;;  patterns are found in this file, items in the User Defined
;;  Hatch Patterns section take precedence over patterns that
;;  appear earlier in the file.
;;
```

TIP On Windows, you can use the Express Tools Super Hatch tool to fill a closed area using a block as the fill pattern. Click Express tab ➢ Draw panel ➢ Super Hatch.

Structure of a Hatch Pattern

Each hatch-pattern definition contains a minimum of two lines of data. The first line defines the pattern's name and allows you to provide an optional description:

```
*pattern_name[, description]
```

When naming or describing a hatch pattern, keep the following in mind:

- The pattern name must be proceeded by an asterisk. The asterisk indicates the start of a new hatch pattern is in a PAT file.
- The pattern name can't include spaces and is limited to a maximum of 31 characters.
- The name should be short and descriptive to make it easy to understand the pattern's intended use.
- Descriptions are optional, but I recommend adding them. If you do add a description, make sure to add a comma after the hatch pattern's name.

The second and subsequent lines of the hatch-pattern definition contain the values that describe how the hatch pattern should look, like this:

angle, *x-origin*,*y-origin*, *x-delta*,*y-delta*

or this:

angle, *x-origin*,*y-origin*, *x-delta*,*y-delta* [,val_1,val_2,val_N,...]

Each pattern-definition line must contain an angle, XY origin, and XY delta. Optionally, the pattern-definition line can contain additional dashes. Other information you need to consider when creating a hatch pattern includes the following:

- Each pattern-definition line is limited to a maximum of 80 characters.
- Each pattern-definition line starts a new line family; the delta offsets are applied to generate parallel lines in an infinite family.
- Text to the right of a semicolon and blank lines are ignored.
- Lines are infinite in length and the dashes in the pattern-definition line are placed on that line.
- A maximum of six dash sequences, including gaps and dots, can be in each pattern-definition line.
- Enter must be pressed after each line, even the last line in a hatch-pattern definition.

Table 9.6 explains the values of a pattern-definition line.

Table 9.6: Pattern-definition line values

Value	Description
angle	Specifies the angle at which the dash sequences of the pattern-definition line should be drawn. The final angle at which the pattern of a hatch object is drawn is the sum of the angle in the hatch pattern, the snapang system variable, and the angle specified during the creation of the hatch object in the drawing.
x-origin	Specifies the X-coordinate value for the start point of the dash sequences in the pattern-definition line. This value is used in combination with the snapbase system variable to control the start point in a drawing when creating a new hatch object.
y-origin	Specifies the Y-coordinate value for the start point of the dash sequences in the pattern-definition line.
x-delta	Specifies the X offset to use between line segments; affects only dashed lines.

TABLE 9.7: Values of the AR-B816 hatch pattern *(CONTINUED)*

`y-delta`	Specifies the Y offset to use between line segments; affects continuous and dashed lines.
`val1,val2,valN,...`	Specifies the dashes, gaps, and dots for noncontinuous lines. These definitions are the same as those used for linetypes: positive value for a dash, negative value for a gap, or a zero for a dot.

With an understanding of the basic structure of a hatch pattern, here is what the definition of the standard AR-B816 hatch pattern looks like from the `acad.pat` file:

```
*AR-B816, 8x16 Block elevation stretcher bond
0,      0,0,     0,8
90,     0,0,     8,8,                          8,-8
```

Table 9.7 explains each value of the AR-B816 hatch pattern.

TABLE 9.7: Values of the AR-B816 hatch pattern

VALUE	DESCRIPTION
`*`	Indicates the start of the hatch pattern.
`AR-B816,`	Hatch pattern's name and separator between the hatch pattern's name and description.
`8x16 Block elevation stretcher bond`	Hatch pattern's description.
`0,`	Lines are drawn with an initial value of 0 .
`0,0,`	Lines start at an initial origin of 0,0.
`0,8`	Lines are offset in the *Y* direction every 8 units.
`90,`	Lines are drawn with an initial value of 90.
`0,0,`	Lines start at an initial origin of 0,0.
`8,8,`	Lines are offset 8 units in the X direction and are staggered 8 units in the Y direction.
`8,-8`	A dash of 8 units is created along the line, followed by a gap of 8 units.

Figure 9.8 shows what each line of the AR-B816 hatch pattern would look if they were separate hatch patterns. You can also see how the offsets, deltas, and dash sequences affect the final results of the hatch pattern.

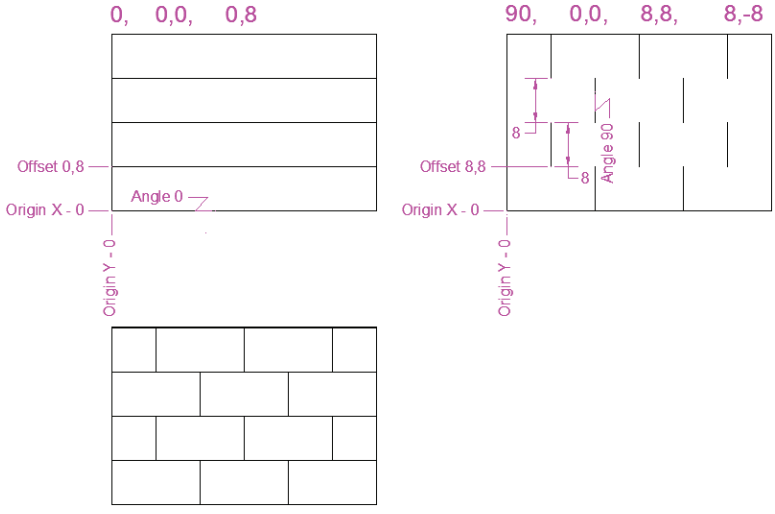


FIGURE 9.8
Breakdown of the AR-B816 hatch pattern

Defining a Custom Hatch Pattern

Hatch-pattern definitions are created and stored in PAT files using an ASCII text editor such as Notepad (Windows) and TextEdit (Mac OS). As I previously mentioned, I recommend storing your custom hatch patterns in PAT files that you create and not those that came with AutoCAD.

The following explains how to create a new PAT file named `diamonds.pat` on Windows:

1. Do one of the following:
   - Click Start button ➤ [All] Programs ➤ Accessories ➤ Notepad (Windows XP and Windows 7).
   - On the Start screen, type **`note`** and click Notepad (Windows 8).
2. In Notepad, click File ➤ Save As.
3. In the Save As dialog box, browse to the `MyCustomFiles` folder you created for the book in the `Documents` (or `My Documents`) folder, or to the location in which you want to create the PAT file.
4. In the File Name text box, type **`diamonds.pat`**.
5. From the Save As Type drop-down list, select All Files (*.*).
6. From the Encoding drop-down list, select ANSI. Click Save.

7. In the editor area, type the following:

```
*Diamonds, Diamond sheeting
60,    0,0,   0.8660,1.5,   1.7321,-1.7321
120,   0,0,   0.8660,-1.5,  1.7321,-1.7321
```

8. Press Enter after the last line in the hatch-pattern definition.
9. Click File ➢ Save.

For AutoCAD on Mac OS, you can use these steps to create a new PAT file named `diamonds.pat`:

1. In the Mac OS Finder, click Go ➢ Applications.
2. In the Finder window, double-click TextEdit.
3. On the Mac OS menu bar, click TextEdit ➢ Preferences.
4. In the Preferences dialog box, click the New Document tab.
5. In the Format section, click Plain Text.
6. Close the Preferences dialog box.
7. Click File ➢ Save.
8. In the Save dialog box, type **`diamonds.pat`** in the Save As text box.
9. Browse to the `MyCustomFiles` folder you created for the book in the `Documents` folder, or to the location in which you want to create the PAT file.
10. From the Plain Text Formatting drop-down list, select Unicode (UTF-8).
11. Check the If No Extension Is Provided, Use *.TXT check box, and then click Save.
12. In the editor area, type the following:

```
*Diamonds, Diamond sheeting
60,    0,0,   0.8660,1.5,   1.7321,-1.7321
120,   0,0,   0.8660,-1.5,  1.7321,-1.7321
```

13. Press Enter after the last line in the hatch-pattern definition.
14. Click File ➢ Save.

Figure 9.9 shows what the Diamonds hatch pattern will look like when used in a drawing.

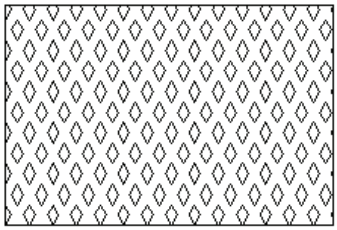


FIGURE 9.9
Diamonds hatch pattern used to fill a rectangle

Using a Custom Hatch Pattern

Custom hatch patterns can be used when creating or modifying a hatch object in a drawing. Before you can use a custom hatch pattern, AutoCAD must be able to locate the file that it is in, unless it was added to `acad.pat` or `acadiso.pat`. Once AutoCAD can locate your PAT file, you can access the hatch pattern just like you would one of the standard patterns that come with the program.

NOTE If you did not create the custom hatch-pattern file in the previous section, you can download the `diamonds.pat` file as part of the sample files for this book. The sample files can be downloaded from `www.sybex.com/go/autocadcustomization`.

Adding Folders with PAT Files to the AutoCAD Support-File Search Paths

The folder that contains your PAT files must be added to the AutoCAD support-file search paths before they can be used with hatch objects. If you have been following the steps in this book, the files should be in the `MyCustomFiles` folder within the `Documents` (or `My Documents`) folder. Otherwise, just make sure you add the folder where the files happen to reside if you placed them elsewhere.

If you are using AutoCAD on Windows, follow these steps:

1. Click the Application button ➢ Options (or at the Command prompt, type **`options`** and press Enter).
2. When the Options dialog box opens, click the Files tab.
3. In the Files tree view, click the plus sign (+) next to the Support File Search Path node. If the folder that contains your PAT file is listed, click OK and close the Options dialog box. Otherwise, continue on.
4. Along the right side of the dialog box, click Add and then click Browse.
5. In the Browse For Folder dialog box, browse to the `MyCustomFiles` folder that you created for this book in the `Documents` (or `My Documents`) folder, or browse to the folder that contains your PAT file if you placed it elsewhere.
6. Select the folder that contains the PAT file and click OK.
7. Click OK to save the changes to the Options dialog box.

If you are using AutoCAD on Mac OS, use these steps:

1. On the menu bar, click AutoCAD <*release*> ➢ Preferences (or at the Command prompt, type **`options`** and press Enter).
2. When the Application Preferences dialog box opens, click the Application tab.
3. In the Files tree view, click the disclosure triangle next to the Support File Search Path node. If the folder that contains your PAT file is listed, click OK and close the Options dialog box. Otherwise, continue on.

4. Near the bottom of the dialog box, click the plus sign (+).
5. In the Open dialog box, browse to the `MyCustomFiles` folder that you created for this book in the `Documents` folder, or browse to the folder that contains your PAT file if you placed it elsewhere.
6. Select the folder that contains the PAT file and click Open.
7. Click OK to save the changes to the Application Preferences dialog box.

Creating Hatch with a Custom Hatch Pattern

After the folder that contains your PAT files has been added to the Support File Search Path node of either the Options dialog box (Windows) or Application Preferences dialog box (Mac OS), you can create a new hatch object using the custom hatch pattern.

NOTE Download the `custom hatch pattern.dwg` file that is part of the sample files for this book. The sample files can be downloaded from `www.sybex.com/go/autocadcustomization`.

The following example explains how to create a hatch object using the ribbon in AutoCAD on Windows:

1. Open the `custom hatch pattern.dwg` file.
2. On the ribbon, click Home tab ➢ Draw panel ➢ Hatch drop-down menu ➢ Hatch (or at the Command prompt, type **`hatch`** and press Enter).
3. Click Hatch Creation tab ➢ Properties panel ➢ Hatch Pattern Style drop-down list ➢ Pattern.
4. Click Hatch Creation tab ➢ Patterns panel ➢ Hatch Pattern gallery ➢ Diamonds (see Figure 9.10). Use the scroll bars along the right side of the Hatch Pattern gallery. Custom hatch patterns are listed near the bottom.

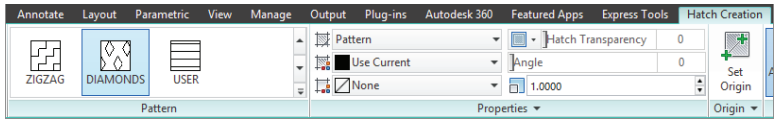


FIGURE 9.10 Diamonds hatch pattern in the Hatch Pattern gallery

5. Position the crosshairs inside the rectangle and click to create the hatch object.
6. Press Enter to end the `hatch` command.

NOTE If you are using a release of AutoCAD on Windows that doesn't support the ribbon or the steps in the example, look up the `hatch` command in the Help system that came with your release of AutoCAD to learn how to create a hatch object.

Use the following to create a hatch object in AutoCAD on Mac OS:

1. Open the `custom hatch pattern.dwg` file.
2. On the menu bar, click Draw ➢ Hatch (or at the Command prompt, type **`hatch`** and press Enter).

3. On the Hatch visor, click the Pattern button (see Figure 9.11). If you click the drop-down menu button, click Open Library.

FIGURE 9.11
Pattern button on the Hatch visor

4. When the Hatch Library palette (see Figure 9.12) opens, click the drop-down list and select Custom.
5. Click Diamonds.
6. Position the crosshairs inside the rectangle and click to create the hatch object.
7. Press Enter to end the `hatch` command.

FIGURE 9.12
Available hatch patterns in the hatch library

NOTE The `MaxHatch` environment variable affects the creation of hatch objects that have dense hatch patterns. If the number of lines that are generated for a hatch object exceed the limit specified by the `MaxHatch` environment variable, a solid hatch is created instead of one made from the hatch pattern. You can use the AutoLISP functions `getenv` and `setenv` to work with an environment variable. The value of the `MaxHatch` environment variable affects only the creation of new hatch objects—and not those already placed in the drawing. The following example changes the `MaxHatch` environment variable to a value of 200000:

```
(setenv "MaxHatch" "200000")
```

The variable name `MaxHatch` is case sensitive; you can't use `maxhatch` or `MAXHATCH`; you must enter it as shown in the sample code.

Chapter 10

Using, Loading, and Managing Custom Files

Throughout this book, I discuss how to define a set of CAD standards that can be applied to all your drawings so they have a consistent look. I show you how to customize the Autodesk® AutoCAD® program on systems running on Windows and Mac OS. Being able to customize the files for AutoCAD on your local workstation is a great start, but in the end you want to make sure that you can deploy or share any customized files with others in your company.

Customizing AutoCAD, as you have read and seen thus far in this book, is a great way to increase your productivity, but you can also download custom programs from the Internet or the Autodesk Exchange store to introduce new functionality and commands. You can also create your own custom programs that improve your company's workflows, but doing so is beyond the scope of this book.

After you spend the time to customize AutoCAD, you will want to make sure you back up your files. How you back up your custom files will vary based on if you are on Windows or Mac OS. Last but not least, recent releases of AutoCAD provide a few utilities that allow you to migrate your customized files to a newer release.

Deploying Your Custom Files

Now that you have taken the time to learn how to customize AutoCAD, you probably can't wait to share what you have learned with others to increase their productivity. Before you start passing files around to everyone in your company, you might want to take a step back and formulate a long-term plan for maintaining any customized files.

Consider the following:

Where will you store and maintain your customized files? The answer to this question will be based on how your environment is configured. Chances are you have a network available to you that most users in your company can access, but you can also use a local drive or a cloud-based file service to share customized files.

Who needs access to your customized files? Usually, customized files are shared within a company, but you might also want to consider sharing your customized files with your subcontractors. This can make it easier to share project files and even complete projects faster. When sharing customized files, consider marking them read-only to avoid accidental changes to the files. Placing files on a network for those in your company is the ideal solution,

because that makes the files more secure and easier to manage. But for those outside your company, a cloud-based file service or an FTP site might be the best choice.

How often do you plan on making changes? The method used to host customized files can impact how many updates you might make on a daily, weekly, or monthly basis. No matter how large or small the changes, you want to create a log of the changes so you can identify any errors that are encountered. This approach also gives your users an idea of what functionality is available to them over time.

Organizing Your Customized Files

As I have mentioned throughout this book, I recommend creating your own or creating a copy of a customizable file over directly modifying the files that come with AutoCAD. Then, once the files are customized, you should place those files in their own location. In the exercises and samples in this book, we used the `MyCustomFolders` folder under the `Documents` (or `My Documents`) folder. This approach has these advantages:

- It lowers the risk of losing your customization during a reset or reinstall of AutoCAD.
- Sharing of customized files is made easier. You can add the folders in which AutoCAD can locate your files as part of the Support File Search Path node in the Options (Windows) or Application Preferences (Mac OS) dialog box.
- Upgrading to a newer release is streamlined, since you know exactly where your customized files reside.
- Backing up customized files is simplified, because you know which files have been customized and where those files reside.

Where to Place Customized Files

I recommend placing all the customized and support files you want to share with others in your organization in a common and shared location instead of manually copying files between workstations. Manually copying files takes extra time and planning that might not be necessary when changes are made to the files. If you place the files on the network, you just need to copy any changed files to a single location, and AutoCAD will load them as long as you have identified the location of the customized files using the Options (Windows) or Application Preferences (Mac OS) dialog box.

No matter whether you store your files on a network or local drive, I recommend defining a folder structure that allows you to use or organize your files by type instead of just placing them all in a single folder. The following is a sample folder structure I have used for my customized files and the environments I have configured for other companies:

```
<company_name>
   <release>
      Blocks
      Plotting
         PC3
         Plot Styles
```

```
Support
   Actions
   Custom Programs
   Icons
   Palettes
Templates
```

The value *<company_name>* in the sample folder structure represents your company's name to help ensure your files are separate from those that come with AutoCAD or even other third-party utilities that might be installed. The value *<release>* represents the AutoCAD release you have installed. If you have more than one release installed, you will want to have a folder structure for each release to avoid potential conflicts between older and newer releases. Customizable files are commonly designed to be forward compatible, but they are not always backward compatible.

Where you create your folder structure depends on the storage option you choose for your customized files. Here are some suggestions based on various storage options:

Network Drive Map a drive letter to the root of the folder structure. This helps ensure that you don't create long folder paths that will cause problems when AutoCAD tries to access the files.

Cloud-Based File-Sharing Service Use the location in which the cloud-based file-sharing service stores local copies of the files that are synchronized from the remote server. For example, Microsoft SkyDrive synchronizes files locally to the `SkyDrive` folder and Dropbox uses the `Dropbox` folder. Both of these folders are located under your user profile.

Removable Drive Create a folder for the customized files on the removable drive at the root level.

Local Drive Create a folder for the customized files on the local drive at the root level, the `ProgramData` folder on Windows, or the `Library` folder on Mac OS.

NOTE Although you could use a folder that is part of a user's Roaming profile for your company's custom files, I recommend not doing that. It requires the user to be logged in to receive any updated files, and it creates duplications of the files under each user's profile. In addition, the files that make up a user's profile should be only the ones that they create or manage themselves.

Maintaining Files Offline

In a perfect world, your customized files would never need to be updated and would only need to be accessible from the company's network. However, Internet connections are not available everywhere and not all connections are fast, stable, or secure. Instead of hoping for an Internet connection, you will want to plan ahead and ensure that remote users have access to the files they need.

For individuals working from home or on a laptop in the field, local access to the most recent files is critical. The same advantages still apply about having your customized files separated

from those that come with AutoCAD, but you need to take an extra step to ensure the user has the latest files. You could use any of the following techniques based on your skill level to maintain a local copy of the customized files that might normally be posted on the network:

- Copying files manually is a low-cost way of getting your files from a network to a local drive. However, it does require someone to remember to do it before they log out and hit the road.
- Using a cloud-based file-sharing service can be a nice and easy-to-set-up solution for taking customized files offline with you. File-sharing platforms like Dropbox, SkyDrive, Google Drive, and others allow you to create a company account and grant read-only access to the files. Since many of the cloud-based services are available on both Windows and Mac OS, they can be useful in mixed environments.
- Creating a batch or shell script that is executed at startup or as part of a Group Policy can be ideal and gives you the most control over where the files are placed and when the synchronization takes place.

There are additional options that are specific to a given operating system, so feel free to experiment and use the solutions that might work best for you to keep your users' local files in sync with those on the network. I have used Robocopy, Offline Files, and Task Scheduler on Windows in the past. You can use rsync or FileSync on Mac OS. These days, I tend to lean toward cloud-based approaches, so I have my customized files available on both my Windows and Mac OS workstations with very little effort.

Customizing the Folders Used by AutoCAD

The folders that AutoCAD looks in for its support files and your customized files are defined as part of the current AutoCAD profile. You can modify the current user profile or create a new one by using the `options` command. AutoCAD doesn't search the subfolders of the folders that are listed as part of the current user profile. Each folder that contains custom program files you want AutoCAD to use must be listed as part of a user profile. I explain managing user profiles in Chapter 4, "Manipulating the Drawing Environment."

NOTE New user profiles can be created only in AutoCAD on Windows.

Based on the locations in which your customized files are stored, you might need to add the folders under the following nodes of the Files tab in the Options dialog box (Windows) or the Application tab in the Application Preferences dialog box (Mac OS). For information about other nodes on the Files or Application tab, see the AutoCAD Help system.

NOTE While you can change the paths listed in the current user profile, be careful. Do not remove the folders that contain the AutoCAD core application and support files. Unexpected results might occur if those folders are removed.

Support File Search Path The Support File Search Path node contains the folders that AutoCAD looks in for its own support files and most of your customized files. Here are some of the file types that should be listed in the folders under this node:

- AutoLISP® (LSP/FAS/VLX) (Windows only)
- Blocks (DWG)
- Compiled shapes (SHX)
- Hatch patterns (PAT)
- Linetype definitions (LIN)
- Managed .NET (DLL) (Windows only)
- ObjectARX® (ARX/CRX/BUNDLE)
- ObjectDBX™ (DBX)
- Program parameter (PGP)
- Scripts (SCR)
- VBA projects (DVB) (Windows only)

Trusted Locations The Trusted Locations node contains the folders of the custom programs that AutoCAD should trust. For files loaded from these locations, AutoCAD does not display a security warning message. I discuss loading custom program files later, in the "Loading a Custom Program" section.

Customization Files The Customization Files node contains the folders where the CUIx or CUI files that define many of the tools in the user interface are stored. You can also define the location that AutoCAD looks in for icons used by your custom tools in the user interface. I discussed customizing the AutoCAD user interface in Chapter 5, "Customizing the AutoCAD User Interface for Windows," and Chapter 6, "Customizing the AutoCAD User Interface for Mac."

Printer Support File Path The Printer Support File Path node contains the folders where the printer configuration (PC3/PCM), printer description (PMP), and plot style (CTB/STB) files are stored. I discussed printer configuration and plot styles in Chapter 1, "Establishing the Foundation for Drawing Standards."

Template Settings The Template Settings node contains the folders where the drawing (DWT) and sheet set/project (DST) template files are stored. I discussed drawing templates in Chapter 1.

Tool Palettes File Locations (Windows Only) The Tool Palettes File Locations node contains the folders where the tool palette (ATC) files are stored for the Tool Palettes window. I discussed tool palettes and the Tool Palettes window in Chapter 7, "Creating Tools and Tool Palettes."

Action Recorder Settings (Windows Only) The Action Recorder Settings node contains the folders where action macro (ACTM) files that AutoCAD can edit and play back are stored. I discussed action macros and the Action Recorder in Chapter 8, "Automating Repetitive Tasks."

The following steps explain how to add the `MyCustomFiles` folder to the Support File Search Path node in the Options dialog box on Windows. This folder is the expected location for the files created as a result of completing the exercises in this book.

1. Click the Application menu button ➢ Options (or at the command prompt, type **`options`** and press Enter).
2. When the Options dialog box opens, click the Files tab.

3. Select the Support File Search Path node and click Add, and then click Browse.
4. In the Browse For Folder dialog box, browse to the `MyCustomFiles` folder that you created for this book in the `Documents` (or `My Documents`) folder, or browse to the folder that contains your customized files.
5. Select the folder that contains your customized files and click OK.
6. Click OK to save the changes to the Options dialog box.

If you are using AutoCAD on Mac OS, use these steps:

1. On the menu bar, click AutoCAD *<release>* ➢ Preferences (or at the command prompt, type **`options`** and press Enter).
2. When the Application Preferences dialog box opens, click the Application tab.
3. Select the Support File Search Path node.
4. Near the bottom of the dialog box, click the plus sign (+).
5. In the Open dialog box, browse to the `MyCustomFiles` folder that you created for this book in the `Documents` folder, or browse to the folder that contains your customized files.
6. Select the folder that contains your customized files and click Open.
7. Click OK to save the changes to the Application Preferences dialog box.

You can change an existing folder in the Options or Application Preferences dialog box by expanding a node and then selecting the node you want to edit. After selecting the folder to edit, click Browse on Windows and select the new folder, or double-click the folder on Mac OS and then select the new folder.

Using and Loading Custom Programs

Everything that I have discussed in this book until now has primarily focused on customizing AutoCAD. While customizing AutoCAD is a great way to increase your productivity, customization does have its limitations when you are trying to automate or simplify complex workflows.

Leveraging one or more of the programming languages that AutoCAD supports with the customization features that you have learned about in this book will help you further increase your productivity. You do not need to learn how to program, but you should be familiar with the programming languages available so you can take advantage of the many custom programs that are available for free (or for a small fee) on the Internet.

What Are Custom Programs?

Custom programs are applications that extend the functionality of AutoCAD through the inclusion of new commands or custom objects, or the extension of a supported programming language. The programming languages that AutoCAD supports allow for a wide range of skills, from basic to advanced. Even if you do not want to learn to be a programmer, having a basic

understanding of a language can be beneficial if you download programs from the Internet. If you do decide to learn a programming language, consider AutoLISP and Visual Basic for Applications (VBA), which are fairly easy to learn with no or little previous experience. Here are the programming languages that AutoCAD supports:

AutoLISP AutoLISP is often the first programming language that a drafter or nonprogrammer learns to extend the functionality of AutoCAD. You can write an AutoLISP program using Notepad on Windows or TextEdit on Mac OS, or the Visual LISP® development environment built into the Windows release of AutoCAD. AutoLISP programs have the file extension `.lsp`. AutoLISP files, which can be compiled, have the file extensions `.fas` and `.vlx`. (VLX files are supported on Windows only.) You can load an AutoLISP file by using the `appload` command or the `load` AutoLISP function.

VBA (Windows Only) VBA is an extension of the Visual Basic (VB) programming language. VBA is an object-oriented programming language that uses the concept of interactive objects defined with properties, methods, and events. When creating VBA programs, you communicate with AutoCAD using the AutoCAD Object Library to create and modify objects in a drawing or the AutoCAD environment. VBA programs are stored in project files with a `.dvb` file extension. DVB files can be loaded and executed using the `vbaman` or `vbarun` command or with the `vl-vbaload` or `vl-vbarun` AutoLISP function.

VB.NET and C# (Windows Only) VB.NET and C# are two of the modern programming languages that AutoCAD supports. Both of these languages require the installation of the .NET Framework, as well as access to specific libraries that are installed with AutoCAD or as part of the ObjectARX software development kit (SDK). Applications created with VB.NET or C# require you to use Microsoft Visual Studio (Express is free, or you can purchase a license for Professional or higher). VB.NET and C# programs must be compiled with the `.dll` file extension and loaded into AutoCAD using the `netload` command.

JavaScript (Windows Only) JavaScript (JS), the most recent programming language that AutoCAD supports, was first introduced with AutoCAD 2014. A JS program can be either a standalone file with the `.js` file extension and loaded with the `webload` command, or part of an HTML file and displayed with the `showhtmlmodalwindow` AutoLISP function. You can create a JS or HTML file using Notepad or a specialized editor.

C++ and Objective-C C++ (Windows) and Objective-C (Mac OS) are two programming languages that are based on the C programming language. Both of these languages require you to download and obtain the ObjectARX SDK, which contains the libraries required to communicate with AutoCAD. An ObjectARX application must be compiled before it can be loaded into AutoCAD. If you want to develop applications with C++, you need to install Microsoft Visual Studio. (Again, Express is free, or you can purchase a license for Professional or higher.) If you are developing applications with Objective-C, you will need to install Xcode. Both languages support the development of graphical user interfaces with MFC (Windows) or Cocoa (Mac OS). ObjectARX programs must be compiled with the `.arx` (Windows) or `.bundle` (Mac OS) file extension. ARX and BUNDLE files can be loaded into AutoCAD using the `appload` command or with the `arxload` AutoLISP function.

NOTE You can download the ObjectARX SDK (which you can use to develop a program using VB.NET, C#, C++, or Objective-C) from `www.objectarx.com`.

Loading a Custom Program

As I previously mentioned, AutoCAD supports a variety of programming languages and each has its own method of being loaded into AutoCAD. The most common types of custom files that you will encounter are AutoLISP (LSP) and ObjectARX (ARX/BUNDLE). Understanding how to load an LSP, ARX, or BUNDLE file allows you to take advantage of the many free routines that others have created and made available for download from the Internet over the past two decades.

Starting with AutoCAD 2013 Service Pack 1 and AutoCAD 2014, Autodesk implemented an additional layer of security that you should be aware of that restricts the loading and execution of custom programs without some user interaction. Prior to those two releases, there were no restrictions on loading and executing custom programs. The following list explains the settings and options that affect the loading and execution of custom programs in AutoCAD 2013 and AutoCAD 2014:

AutoCAD 2013 with Service Pack 1 (Windows Only) The system variables and the command-line switch that affect the loading of custom programs in AutoCAD 2013 are as follows:

- `autoload`, which specifies whether AutoCAD can automatically load custom files with specific names at startup
- `autoloadpath`, which specifies the folders from which AutoCAD can automatically load custom files with specific names at startup
- `lispenabled`, which indicates whether AutoLISP has been disabled in the current session
- `/nolisp`, which disables the execution of AutoLISP (LSP/FAS/VLX) (Windows only) files in the current session

AutoCAD 2014 The system variables and command-line switches that affect the loading of custom programs in AutoCAD 2014 are as follows:

- `secureload`, which specifies whether AutoCAD can load executable files from any folder with or without a warning, or whether it can load files only from the folders specified by the `trustedpaths` system variable.
- `trustedpaths`, which specifies the folders from which AutoLISP, VBA, Managed .NET, and ObjectARX files can be loaded without user interaction. VBA and Managed .NET are supported on Windows only.
- `trusteddomains`, which specifies the domains and URLs from which JS files can be loaded without user interaction (Windows only).
- `safemode`, which indicates whether executable files can be loaded and executed in the current session.
- `/safemode`, which disables the execution of executable files in the current session (Windows only).
- `-safemode`, which disables the execution of executable files in the current session (Mac OS only).

The following steps explain how to load an AutoLISP or ObjectARX application:

1. Do one of the following:
   - On the ribbon, click the Manage tab ➢ Applications panel ➢ Load Application (Windows).

- On the menu bar, click Tools ➤ Load Application (Mac OS).
- At the command prompt, type **appload** and press Enter (Windows and Mac OS).

2. When the Load/Unload Application dialog box (see Figure 10.1) opens, browse to and select the AutoLISP or ObjectARX file you want to load. Click Load.

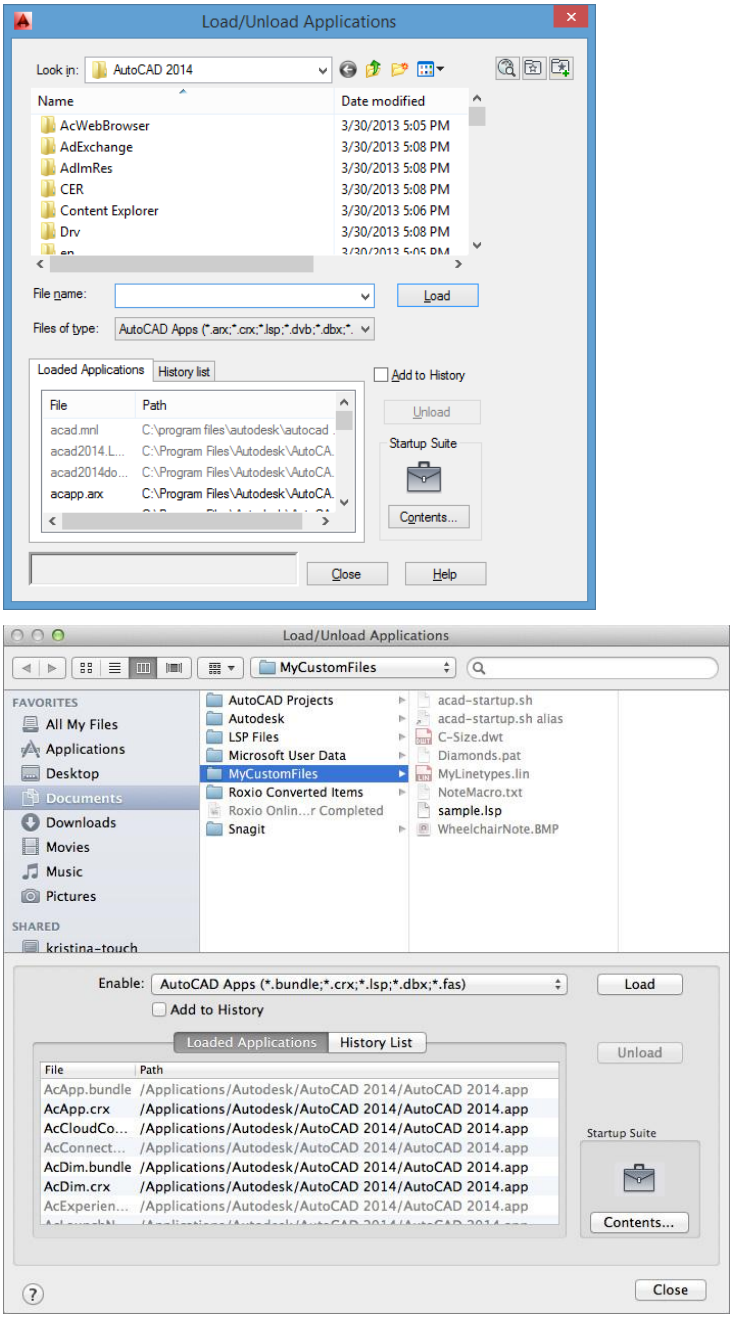


FIGURE 10.1
Loading a custom program

3. If the File Loading - Security Concerns message box is displayed, click Load. See the topics on the `secureload` and `trustedpaths` system variables in the AutoCAD 2014 Help system for information on how to avoid this message each time you load a custom program.
4. Click Close to return to the drawing area.
5. Execute any new command that is made available after the custom program file is loaded.

You can use the Startup Suite section of the Load/Unload Applications dialog box to automatically load custom program files each time AutoCAD starts. For information on using the Startup Suite, see the `appload` command in the AutoCAD Help system.

Locating Custom Programs on the Internet

Custom programs can be found on the Internet by performing a search using your favorite search engine. You can find some custom program files by doing a search on a combination of the keywords `autolisp`, `objectarx`, `programs`, `files`, `sample`, `download`, and `free`. An example keyword search might be `autolisp files`; the results in Bing are shown in Figure 10.2.

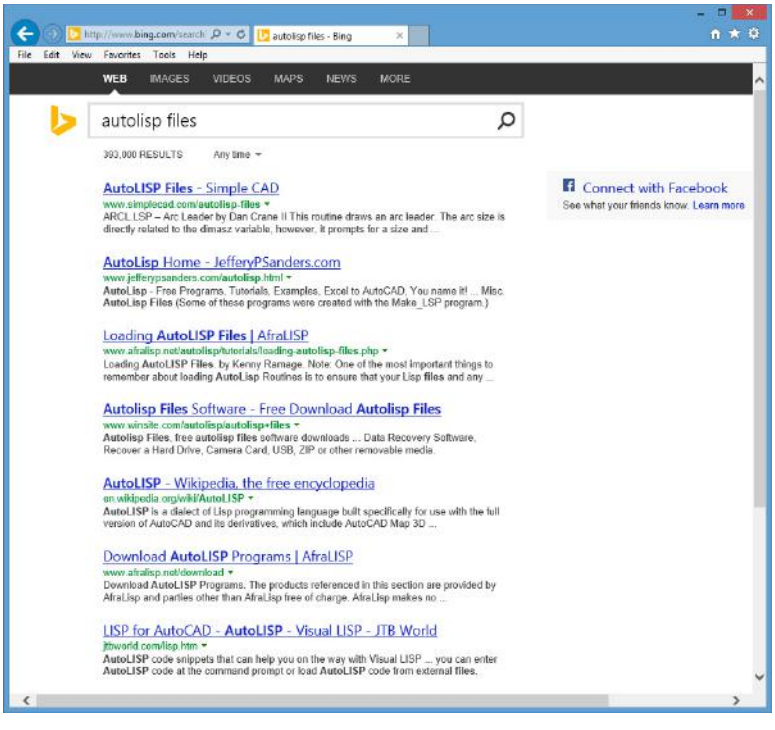


FIGURE 10.2 Internet search results on AutoLISP files

WARNING With any files that you download from the Internet, make sure you scan the files with virus software. Also, download files from a reputable website whenever possible. If you are unsure about a website that contains free custom program files for AutoCAD, check whether someone on the Autodesk (`forums.autodesk.com`) or AUGI (`forums.augi.com`) forums has downloaded files from the website.

The following lists my website (HyperPics) and other reputable sites that offer free custom programs that can be used with AutoCAD:

- AfraLISP: `www.afralisp.net`
- DotSoft: `www.dotsoft.com`
- HyperPics: `www.hyperpics.com/customization/`
- JTB World: `www.jtbworld.com`
- ManuSoft: `www.manusoft.com`

If you are using AutoCAD 2013 or AutoCAD 2014 on Windows, you can also download and install custom programs through the Autodesk Exchange App Store (`http://apps.exchange.autodesk.com/`). The Autodesk Exchange App Store contains a variety of apps not only for AutoCAD, but also for AutoCAD-based vertical applications and other Autodesk products. You can find both free and paid for apps in the store.

Backing Up and Migrating Customization

Backing up your custom settings and files is often one of the last tasks you perform after customizing AutoCAD, but it is also an important one if something were to go wrong. I recommend backing up frequently—that comes out of habit after forgetting to back up a few times over the years and having to redo several days of customization work. While backing up files is often one of the last tasks you perform, migrating is one of the first tasks you will perform when upgrading to a new release.

The most basic way to back up and migrate your files on either Windows or Mac OS is to manually copy your files between folders. The following lists the folders that AutoCAD uses by default to store the customized files it can make changes to:

- `%AppData%\Autodesk\<product_name>\<release>\<language>` (Windows)
- `%LOCALAPPDATA%\Autodesk\<product_name>\<release>\<language>` (Windows)
- `~Library/Application Support/Autodesk/roaming/<product_name>/<release>/<language>` (Mac OS)
- `~Library/Application Support/Autodesk/local/<product_name>/<release>/<language>` (Mac OS)

As I discussed earlier in this chapter, I strongly recommend storing your customization in separate folders and files from those that AutoCAD creates after installation. If you follow my recommendation, be sure to back up the folders and files in those locations in addition to the ones that AutoCAD updates.

If you are using AutoCAD on Windows, there are some additional utilities available for backing up custom settings and files, synchronizing custom settings with your Autodesk 360 account, and helping to migrate custom settings to a new release. These utilities include the following:

Export AutoCAD <*release*> Settings The Export AutoCAD <*release*> Settings option on the Windows Start menu or Start screen allows you to export the custom settings and files of AutoCAD to a ZIP file.

Import AutoCAD <*release*> Settings The Import AutoCAD <*release*> Settings option on the Windows Start menu or Start screen allows you to import the custom settings and files that were previously exported using Export AutoCAD <*release*> Settings.

Synchronize Custom Settings The Synchronize Custom Settings feature allows you to back up your custom settings and files to your Autodesk 360 account. These settings can then be restored on another workstation or on the same workstation they were backed up from. You can control whether you want to use this feature, and the types of settings and files that can be backed up, on the Online tab of the Options dialog box (`options` command).

Migrate From A Previous Release The Migrate From A Previous Release option on the Windows Start menu or Start screen allows you to migrate your custom settings and files from a previous release to a newer release. For example, you can migrate your settings from AutoCAD 2012 to AutoCAD 2014. Both releases need to be installed in order to migrate the custom settings and files.

Customize User Interface, Transfer Tab The Transfer tab of the Customize User Interface (CUI) Editor allows you to transfer user-interface elements between two different CUIx files. This can be helpful if you want to migrate only some of the elements in your user interface to a new release. I discussed the CUI Editor in Chapter 5.

For more information on these utilities, see the AutoCAD Help system.

TIP For additional information on issues that could affect migrating custom settings and files, or using custom program files from an earlier release, see the CAD Administration Guide in the AutoCAD Help system (`docs.autodesk.com/ACD/2014/ENU/`).

Index

Note to the Reader: Throughout this index **boldfaced** page numbers indicate primary discussions of a topic. *Italicized* page numbers indicate illustrations.

B

C

D

I

J

L

M

N

Q

R

S

T

U

V

W

X

Y